U.S. DEPARTMENT OF COMMERCE
National Technical Information Service

AD-A032 034

Fracture Mechanics Analysis of Centered & Offset Fastener Holes in Stiffened & Unstiffened Panels under Uniform Tension

Massachusetts Inst of Tech Cambridge

Apr 76

322103

ADA032034

AFFDL-TR-75-70

FRACTURE MECHANICS ANALYSIS OF CENTERED AND OFFSET FASTENER HOLES IN STIFFENED AND UNSTIFFENED PANELS UNDER UNIFORM TENSION

AEROELASTIC AND STRUCTURES RESEARCH LABORATORY
DEPARTMENT OF AERONAUTICS AND ASTRONAUTICS
MASSACHUSETTS INSTITUTE OF TECHNOLOGY
CAMBRIDGE, MASSACHUSETTS 02139

APRIL 1976

TECHNICAL REPORT AFFDL-TR-75-70
INTERIM REPORT FOR PERIOD OCTOBER 1974 - FEBRUARY 1975

Prepared For
AIR FORCE FLIGHT DYNAMICS LABORATORY
AIR FORCE SYSTEMS COMMAND
WRIGHT-PATTERSON AIR FORCE BASE, OHIO 45433

NOTICE

When Government drawings, specifications, or other data are used for any purpose other than in connection with a definitely related Government procurement operation, the United States Government thereby incurs no responsibility nor any obligation whatsoever; and the fact that the government may have formulated, furnished, or in any way supplied the said drawings, specifications, or other data is not to be regarded by implication or otherwise as in any manner licensing the holder or any other person or corporation, or conveying any rights or permission to manufacture, use, or sell any patented invention that may in any way be related thereto.

This report has been reviewed by the Information Office (OI) and is releasable to the National Technical Information Service (NTIS). At NTIS, it will be available to the general public including foreign nations.

This technical report has been reviewed and is approved for publication.

James L. Rudd

JAMES L. RUDD
Project Engineer

Robert M. Bader

ROBERT M. BADER, Chief
Structural Integrity Branch
Structural Mechanics Division

FOR THE COMMANDER

Howard L. Farmer

HOWARD L. FARMER, Colonel, USAF
Chief, Structural Mechanics Division

Copies of this report should not be returned unless return is required by security considerations, contractual obligations, or notice on a specific document.

AIR FORCE - 29 OCTOBER 76 - 300

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER AFFDL-TR-75-70	2. GOVT ACCESSION NO. NA	3. RECIPIENT'S CATALOG NUMBER NA
4. TITLE (and Subtitle) FRACTURE MECHANICS ANALYSIS OF CENTERED AND OFFSET FASTENER HOLES IN STIFFENED AND UNSTIFFENED PANELS UNDER UNIFORM TENSION		5. TYPE OF REPORT & PERIOD COVERED INTERIM Oct. 1974-Feb. 1975
		6. PERFORMING ORG. REPORT NUMBER ASRL TR 177-2
7. AUTHOR(s) George Stalk, Oscar Orringer		8. CONTRACT OR GRANT NUMBER(s) F33615-74-C-3063
9. PERFORMING ORGANIZATION NAME AND ADDRESS Aeroelastic and Structures Research Lab. Department of Aeronautics and Astronautics Massachusetts Institute of Technology Cambridge, Massachusetts 02139		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS 13670315
11. CONTROLLING OFFICE NAME AND ADDRESS Air Force Flight Dynamics Laboratory Air Force Systems Command Wright-Patterson AFB, Ohio 45433		12. REPORT DATE APRIL 1976
		13. NUMBER OF PAGES 185
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office) Same		15. SECURITY CLASS. (of this report) Unclassified
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE NA

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

Same

18. SUPPLEMENTARY NOTES

None

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Fracture Mechanics
Stress Intensity Factors
Finite Element Analysis
Hybrid Elements
Aircraft Structural Integrity
Stress Analysis
Computer Structural Analysis

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

This report documents a finite-element analysis procedure for computation of Mode I and Mode II stress intensity factors associated with one or two sharp cracks emanating from a fastener hole in a panel under uniform tension. The fastener hole may be offset from the panel centerline, and one or both edges of the panel may be integrally stiffened. The formulation of a special assumed-stress hybrid element for the region near the fastener

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

hole is presented. A detailed description of the programming covers the physical problem, modelling, program flow and options, input/output conventions, execution times, and limitations which must be observed. Results are presented for performance tests of the special element, and for some example analyses of cracked panels.

FOREWORD

The developments documented in this report were carried out at the Aeroelastic and Structures Research Laboratory, Massachusetts Institute of Technology, Cambridge, Massachusetts 02139, under Contract No. F33615-74-C-3063 (Project 1367, Task 136703) from the U.S. Air Force Flight Dynamics Laboratory. Mr. James L. Rudd (AFFDL/FBE) served as technical monitor. The authors gratefully acknowledge the many contributions by Mrs. Susan E. French of the Aeroelastic and Structures Research Laboratory. Mrs. French was involved with the detailed programming aspects of the work throughout the project. This report is the second in a series, covering research conducted during October 1974-February 1975, and was submitted for technical review in May 1975. The other reports in this series are AFFDL-TR-75-51 (Fracture Mechanics Analysis of an Attachment Lug), AFFDL-TR 75-71, (Fracture Mechanics Analysis of Single and Double Rows of Fastener Holes Loaded in Bearing), and AFFDL-TR-76-12 (Numerical Computation of Stress Intensity Factors for Aircraft Structural Details by the Finite Element Method). The contractor's report number is ASRL TR 177-2.

TABLE OF CONTENTS

TABLE OF CONTENTS (Concluded)

LIST OF ILLUSTRATIONS

LIST OF ILLUSTRATIONS (Continued)

LIST OF ILLUSTRATIONS (Concluded)

LIST OF TABLES

Section 1

INTRODUCTION

This is the second of a series of reports on the development of finite-element analysis procedures for computation of linear elastic stress intensity factors associated with cracks in common aircraft structural details. Finite-element analysis serves the important purpose of providing K_I and K_{II} solutions for situations in which the boundary geometries are too complicated to permit convenient treatment by the classical methods.

The first report in the series [1] summarized the formulation of two basic building blocks which are used in the analyses: the well-known bilinear isoparametric quadrilateral and the PCRK59 assumed-stress hybrid crack-containing elements. The former element has been widely accepted and appears in most general-purpose finite-element programs. The latter element was originally developed at ASRL in 1972 [2], and has subsequently been put through extensive performance tests to characterize its behavior as a function of shape distortions [3].

Earlier work [4, 5] has demonstrated that the assumed-stress hybrid method permits accurate computation of stress intensity factors, while preserving Matrix Displacement Method conventions in the global programming environment (data input, assembly, matrix solution procedure, etc.). In the present work, the hybrid method is used again to formulate another special-

purpose element for modelling the region around a fastener hole. The new element is combined with the quadrilateral and PCRK59 to represent a skin tension panel containing an open fastener hole. The ASRL FEABL program [6] is again used for global analysis. Additional features belonging to FEABL which have not yet been formally documented are employed in the present analysis, and are outlined briefly in this report. These features pertain to the technique of substructuring by a Gauss elimination algorithm.

Section 2

NEW DEVELOPMENTS

Modularization of the finite-element model of a structure becomes as important as modularization of the basic program when the application is computation of stress intensity factors. The objective is to reduce unnecessary arithmetic operations and computer printout to a minimum. In particular, the stresses and displacements are of little interest (and of no interest at all at any distance away from the crack tip) except for verifications of accuracy during development. The substructuring technique, used for many years in airframe analysis programs*, provides the required modularity. Subroutines for substructuring were developed as add-ons to FEABL as part of an unrelated project** and have been used in the present work. The substructuring algorithm and the functions and conventions of the add-on subroutines are discussed briefly in this section to provide a complete picture of the panel analysis program.

The so-called "near-field" region around a fastener hole forms one such substructure module. The first attempt to model this region with conventional quadrilateral elements resulted in

*For example, the Boeing ASTRA program.

**Sponsored by the Xerox Corporation (Dr. T. C. Soong, technical monitor).

severe mesh distortion, numerous unwanted degrees of freedom, and solution accuracy problems. In consequence, the hybrid method was called upon, resulting in the HOLE1 special-purpose element for this module. The history of these investigations is traced in Subsections 2.3 and 2.4.

2.1 Substructuring

Figure 1 illustrates the finite-element model of a square panel which contains a single fastener hole offset from the panel centerline. A detail in the lower part of the figure shows how a PCRK59 element might be inserted to represent a small crack emanating from the hole. It is evident that the complete global solution (displacements at each of the many nodes in the model) constitutes unwanted information if the analysis is to be repeated while the crack location is varied parametrically around the fastener hole.

A computationally more efficient procedure is obtained by subdividing the model into far-field and near-field substructures, as shown in Fig. 2. The broken line in Fig. 2 corresponds to the edges ABCD in Fig. 1. If the element stiffnesses, prescribed nodal forces and nodal displacements are first assembled for the far-field substructure, the resulting equation system may be partially solved. In effect, all of the information pertaining to the far field substructure is transferred to the inter-substructure boundary ABCD. This information may then be treated as a "super-element", which can be assembled with a succession

of near-field substructures containing a crack in different locations. As a result, many fewer degrees of freedom are processed in the parametric stress intensity computations.

The substructuring process may be represented formally as follows. Let the assembled equation system $\underset{\sim}{K}\underset{\sim}{q} = \hat{\underset{\sim}{Q}}$ for a substructure be partitioned into:

$$\begin{bmatrix} \underset{\sim}{K}_{II} & \underset{\sim}{K}_{IB} \\ \underset{\sim}{K}_{BI} & \underset{\sim}{K}_{BB} \end{bmatrix} \begin{Bmatrix} \underset{\sim}{q}_I \\ \underset{\sim}{q}_B \end{Bmatrix} = \begin{Bmatrix} \hat{\underset{\sim}{Q}}_I \\ \hat{\underset{\sim}{Q}}_B \end{Bmatrix} \tag{1}$$

where

$\underset{\sim}{K}$ = stiffness coefficients

$\underset{\sim}{q}$ = (unknown) nodal displacements

$\hat{\underset{\sim}{Q}}$ = prescribed nodal forces

and where $\underset{\sim}{K}_{BI} = \underset{\sim}{K}^T_{IB}$. Subscripts I and B refer to nodes and degrees of freedom which are considered respectively as "interior" (to be eliminated) and "boundary" (to be retained for subsequent assembly along the inter-substructure boundary). The interior degrees of freedom are formally eliminated by solving the first of Eqs. 1 and substituting into the second:

$$\underset{\sim}{K}_{II}\underset{\sim}{q}_I + \underset{\sim}{K}_{IB}\underset{\sim}{q}_B = \hat{\underset{\sim}{Q}}_I$$

$$\underset{\sim}{q}_I = \underset{\sim}{K}^{-1}_{II}\hat{\underset{\sim}{Q}}_I - \underset{\sim}{K}^{-1}_{II}\underset{\sim}{K}_{IB}\underset{\sim}{q}_B \tag{2}$$

$$\underset{\sim}{K}_{BI}\left(\underset{\sim}{K}^{-1}_{II}\hat{\underset{\sim}{Q}}_I - \underset{\sim}{K}^{-1}_{II}\underset{\sim}{K}_{IB}\underset{\sim}{q}_B\right) + \underset{\sim}{K}_{BB}\underset{\sim}{q}_B = \hat{\underset{\sim}{Q}}_B \tag{3}$$

A rearrangement of the terms in Eq. 3 leads immediately to:

$$\underset{\sim}{K}_{BB}^{(c)} \underset{\sim}{q}_B = \hat{\underset{\sim}{Q}}_{BB}^{(c)} \tag{4}$$

where $\underset{\sim}{K}_{BB}^{(c)}$, $\hat{\underset{\sim}{Q}}_{B}^{(c)}$ are the statically condensed stiffnesses and forces, given by:

$$\underset{\sim}{K}_{BB}^{(c)} = \underset{\sim}{K}_{BB} - \underset{\sim}{K}_{BI} \underset{\sim}{K}_{II}^{-1} \underset{\sim}{K}_{IB} \tag{5}$$

$$\hat{\underset{\sim}{Q}}_{B}^{(c)} = \hat{\underset{\sim}{Q}}_{B} - \underset{\sim}{K}_{BI} \underset{\sim}{K}_{II}^{-1} \hat{\underset{\sim}{Q}}_{I} \tag{6}$$

It is apparent from Eqs. 5 and 6 that the substructuring process may be realized as a computing algorithm independent of the unknowns $\underset{\sim}{q}_I$, $\underset{\sim}{q}_B$. In practice, the process may be programmed to eliminate one degree of freedom (one equation) from the system at a time, so that inversion of $\underset{\sim}{K}_{II}$ is avoided. The entire algorithm then consists of as many passes through the assembled equations as there are degrees of freedom to be eliminated, a procedure quite similar to the Gauss-Cholesky factoring methods normally used to solve the whole equation system [6].

Substructuring places an additional burden on the program user when the global software stores $\underset{\sim}{K}$ as a band-matrix. It can be shown by tracing the details that the elimination algorithm inflates the stiffness matrix bandwidth if $\underset{\sim}{q}_I$ and $\underset{\sim}{q}_B$ are arbitrarily

interspersed. However, if $\underset{\sim}{q}_I$ always appear as the first degrees of freedom in the global numbering sequence, the bandwidth of $\underset{\sim}{K}$ is not affected by elimination. This restriction has been placed on the FEABL add-on subroutines, and it requires some care in the choice of global numbering sequence and substructure boundary locations on the user's part.

The foregoing derivation of the substructuring algorithm assumes that all the prescribed quantities are nodal forces $\underset{\sim}{\hat{Q}}$, while all of the displacements $\underset{\sim}{q}$ are unknowns. However, if restraints are applied to the substructure such that a subset $\underset{\sim}{\hat{q}}_{I_o}$ of the interior degrees of freedom $\underset{\sim}{q}_I$ are prescribed, Eqs. 2 through 6 may still be used. The only difference is that the rows and columns of $\underset{\sim}{K}$ corresponding to $\underset{\sim}{\hat{q}}_{I_o}$ are decoupled from the equation system*, while the right hand side of Eqs. 1 is replaced by:

$$\left\{ \underset{\sim}{\hat{Q}}_I \;\vdots\; \underset{\sim}{\hat{Q}}_B \right\} = \left\{ \underset{\sim}{\hat{Q}}_I \text{ or } \underset{\sim}{\hat{q}}_{I_o} \;\vdots\; \underset{\sim}{\hat{Q}}_B - \underset{\sim}{K}_{BI_o} \underset{\sim}{\hat{q}}_{I_o} \right\} \tag{7}$$

2.2 Implementation of Substructuring in FEABL

Three add-on subroutines have been programmed and verified for the gl al computation tasks associated with substructuring. Subroutine STACON executes the Gauss elimination algorithm on a band-matrix which has been assembled and stored in accordance with

*See Ref. 6, Subsection 3.2.5.

the existing FEABL conventions. A normal MAIN program is written to control the FEABL software up to the point at which $\underset{\sim}{K}$ would be factored in a conventional analysis. However, instead of factoring the instructions:

```
ISGN =
CALL STACON(ISGN,NID,RNAME,INAME)
```

is programmed, where

ISGN = Matrix condition control parameter

NID = Total number of interior degrees of freedom

RNAME,INAME = Real (floating point) and integer (fixed point) variable names assigned to the FEABL DATA vector

The control parameter ISGN must be set to +1, -1 or 0 before STACON is called. A value of +1 aborts the run if $\underset{\sim}{K}$ is not a positive-definite matrix. A value of -1 aborts the run if $\underset{\sim}{K}$ is singular. A value of 0 always results in return to the MAIN program with ISGN reset according to the actual matrix condition encountered: positive-definite (+1), non-positive but nonsingular (negative integer), or singular (0). The positive-definite option is used in conventional finite-element analyses. The nonsingular option is used in special cases, e.g., Hermann's principle for which ideally incompressible elements include volumetric constraints in the assembled equation system. The option ISGN=0 may be used generally, in programs for which the user wishes to retain control for debugging or other purposes when an error condition is encountered. Subroutine STACON carries out the elimination process in place,

as shown schematically in Fig. 3. After execution, the statically condensed force vector appears in INAME(IQ+NID) to INAME(LQ) and the statically condensed stiffness coefficients appear in INAME(J) to INAME(LK), where:

$$J=IK+INAME(IKØUNT+NID)+NID+1 \tag{8}$$*

The shaded areas shown in Fig. 3 contain the back-substitution information $\underset{\sim}{K}_{II}^{-1}\ \hat{\underset{\sim}{Q}}_I$ and $\underset{\sim}{K}_{II}^{-1}\ \underset{\sim}{K}_{IB}$, which may be used to solve for $\underset{\sim}{q}_I$ (Eq. 2) after $\underset{\sim}{q}_B$ has been computed.

The statically condensed stiffnesses and forces may simply be transferred to another storage area and treated thereafter as a "super-element" if desired. A general programming sequence to execute the transfer is given in Appendix A. If this is done, the substructure may later be assembled into a complete structure with subroutine ASEMBL or subroutine ASMLTV. A list of local-to-global degree-of-freedom connections must be given for the "super-element", just as is done for any ordinary element. However, while the numbering convention for an ordinary element is fixed by the manner in which the element subroutine has been programmed, the convention for a "super-element" is determined by the user's sequence of global numbering for assembly of the substructure. For example, if the far-field substructure in Fig. 2 has been

*IKØUNT is another address control parameter. See Ref. 6, subsections 2.1.3, 2.2.2 and 2.2.6.

numbered such that the boundary degrees of freedom are globally assigned in the order A→B→C→D→A, then the connections of the far-field "super-element" to the near-field portion of the model must be given in that order.

A second option available to the user is to leave the statically condensed substructure in place and to employ a second DATA vector for assembly and solution of the complete structure. Add-on subroutine ASMSUB has been programmed for this option. Subroutine ASMSUB performs a function similar to the conventional assembly routines (ASEMBL,ASMLTV), but is capable of assembling $\underset{\sim}{K}_{BB}^{(c)}$, $\hat{\underset{\sim}{Q}}_{B}^{(c)}$ directly from one DATA vector into another. Since two DATA vectors must be manipulated simultaneously, extra address control parameters are required. The beginning of the program might appear as follows:

```
DIMENSIØN RNAME1(xxx), INAME1(xxx), RNAME2(yyy), INAME2(yyy)
EQUIVALENCE (RNAME1(1), INAME1(1))
EQUIVALENCE (RNAME2(1), INAME2(1))
CØMMØN /SIZE/ NET, NDT
CØMMØN /BEGIN/ ICØN,IKØUNT,ILNZ, IMASTR, IQ, IK
CØMMØN /END/ LCØN,LKØUNT,LLNZ, LMASTR,LQ, LK
CØMMØN /SIZESS/ NETSS,NDTSS,NIDSS
CØMMØN /BEGSS/ IBEG(6)
CØMMØN /ENDSS/ LEND(6)
.
.
.
```

Now suppose that a substructure is assembled and statically

condensed in vector (RNAME1,INAME1):

```
      .
      .
      .
      ISGN=1
      CALL STACON(ISGN,NID,RNAME1,INAME1)
```

At this point, the substructure's control information must be cleared from the labelled COMMON areas to permit formation of the complete structure. The information is transferred to the extra areas:

```
      NETSS=NET
      NDTSS=NDT
      NIDSS=NID
      IBEG(1)=ICON
      .
      .
      .
      LEND(6)=LK
      .
      .
      .
```

The program continues with a conventional generation of size, connections, assembly, etc. for the complete structure in vector (RNAME2,INAME2) until the substructure is to be assembled. At this point, the user simply programs:

```
      .
      .
      .
      CALL ASMSUB(LNUM, RNAME1,INAME1,RNAME2,INAME2)
```

.
.
.

where LNUM represents the element number assigned to the substructure, considered as a "super-element" in the complete structure.

There are some cases for which the solution q_I may be required in a particular substructure. Subroutine QBACK has been programmed as the third add-on to execute the required back-substitution solution for q_I. Subroutine QBACK is designed to work with ASMSUB in the multiple DATA vector environment. Suppose that the above example is continued to the point at which assembly and constraint of the structure equation system are complete. Factoring and global solution now follow:

```
.
.
.

CALL FACT(ISGN,RNAME2, INAME2)
CALL SIMULQ(ENERGY, RNAME2, INAME2)
```

At this point, vector (RNAME2, INAME2) contains the solution q for the complete structure. Part of q is, in fact, the boundary displacement solution q_B for the substructure. An interior solution may be obtained by continuing with:

```
CALL QBACK(LNUM,RNAME1,INAME1,RNAME2,INAME2)

.
.
.
```

After execution of QBACK, the substructure global solution $(q_I \vdots q_B)$ appears in vector (RNAME1,INAME1). The solution will also be

printed by subroutine QBACK.

2.3 Experiment with Conventional Mesh

The first attempt to model the panel and fastener hole was made with conventional quadrilaterals and a 5-node hybrid mesh-expander element. Experience with analysis of an attachment lug detail [1] indicated that at least 24 quadrilaterals should be placed around the hole to obtain accurate solutions. One then faces the following choices for the panel:

1. Mesh expansion in the near-field region to reduce the number of divisions below 24 on the near/far-field boundary (ABCD in Figs. 1 and 2).
2. Acceptance of 24 divisions on the near/far-field boundary and continuation of this detail into the far-field region.
3. Acceptance of 24 divisions on the boundary, with some mesh expansion immediately outside the near-field region.

The first choice was judged to be unlikely to give accurate computed displacements near the hole because of the severe shape distortions which the mesh-expander elements would have caused. The second choice was deemed to be unacceptable in that too many unwanted degrees of freedom would be placed in the far-field region, making either one-stage solution or substructuring computationally inefficient. The third choice was consequently selected as the best compromise between computational efficiency and solution accuracy near the hole.

Figure 4 illustrates a typical mesh for a panel with the fastener hole centered. Extra detail is kept in the far-field

region to the right of the hole in order to permit the hole to be offset continuously toward the right edge of the panel while avoiding extreme aspect ratios in the far-field elements. Tests with this model quickly showed that the asymmetric mesh grading led to highly asymmetric behavior of the computed solutions. Figure 5 illustrates a typical example. Computed horizontal displacement of the left and right edges of the panel are shown for the case of uniform vertical tension and the fastener hole centered. The inaccuracy of the model was judged to have been caused by the combination of asymmetric mesh grading, aspect ratio effects, and shape distortions in the near-field region.

The conclusion was that conventional finite-element models could be made to work for the panel structure only with very fine detail, and therefor at an unacceptably high computing cost. Lest it be thought that too much accuracy has been demanded from the analysis, we remind the reader that more than an engineering solution is required to achieve the ultimate goal. In considering the accuracy of the K_I and K_{II} solutions, it has been shown [1, 3] that shape distortions of the PCRK59 element alone can account for as much as 3 to 5 percent error. Furthermore, direct verification of accuracy by comparison with independent methods is possible only for K_I with cracks emanating horizontally from the fastener hole. Hence, a finite-element model capable of computing highly accurate displacements near the fastener hole is required to achieve engineering accuracy in the final answer. As a result, the assumed-stress hybrid method was investigated as a possible

alternate way of handling the transition from the circular hole geometry to the cartesian geometry of the near/far-field boundary.

2.4 Creation of a Hybrid Fastener Hole Element

The energy principles on which hybrid elements are based have been reviewed previously [1]. The energy principle Π_1 was chosen for the present investigation, giving the general form of an element stiffness matrix as:

$$\underset{\sim}{k} = \underset{\sim}{G}^T \underset{\sim}{H}^{-1} \underset{\sim}{G} \tag{9}$$

where

$$\underset{\sim}{G} = t \int_{\partial A} (\underset{\sim}{N}\underset{\sim}{P})^T \underset{\sim}{L} \, dA \tag{10}$$

$$\underset{\sim}{H} = t \int_{A} \underset{\sim}{P}^T \underset{\sim}{S} \underset{\sim}{P} \, dA \tag{11}$$

and where

A = Element area

∂A = Element boundary

$\underset{\sim}{L}$ = Matrix of assumed-displacement interpolation functions defined on ∂A.

$\underset{\sim}{N}$ = Matrix of direction cosines for the outward normal to ∂A.

$\underset{\sim}{P}$ = Matrix of assumed-stress shape functions.

$\underset{\sim}{S}$ = Matrix of elastic compliance constants.

t = Element thickness (assumed to be unity without loss of generality).

Two fundamental properties of hybrid elements make them useful in linear elastic plane stress and plane strain analyses:

1. Great flexibility in choice of element shape is permitted, since $\underset{\sim}{L}$ need be defined only in terms of an arc length coordinate from node to node along ∂A, and since the shape functions $\underset{\sim}{P}$ need only satisfy the continuum equilibrium equations.
2. Many shape functions $\underset{\sim}{P}$ can be chosen to satisfy equilibrium and at the same time allow the element to closely mimic a particular local stress distribution.

Both properties have been used to advantage in the development of an element for the near-field region.

Practical application of the hybrid method requires that the analyst account for known local behavior in the formulation of his element. In the present case, one reasonably expects to see $\sin(2\theta)$ and $\cos(2\theta)$ components in the stress distribution near a fastener hole in a panel under tension, as well as terms independent of θ. Second, one should expect to see the near/far-field boundary deform in the manner shown schematically in Fig. 6. Simulation of this behavior requires that the element have at least three nodes (corners and mid-edge) along its near/far-field boundary edge. Third, the element must be kinematically stable, since it will be placed in the mesh so as to separate two conventionally modelled regions. Kinematic stability is assured if:

$$m \geqslant n - 3 \qquad (12)$$

for plane stress and plane strain elements, where

m = Total number of assumed-stress shape functions.

n = Total number of nodal degrees of freedom (displacements) in the element.

Finally, at least 20 nodes should be provided along the circular inner boundary, which may serve either as the fastener hole boundary or as an interface to which quadrilaterals and PCRK59 elements may be coupled.

The kinematic stability condition results in a quick determination that modelling of the entire near-field region with a single element would be impractical. For example, assumption of the minimum number of nodes (8 along the outer, 20 along the inner boundary) results in a 56-degree of freedom element which requires $m \geq 53$ assumed-stress shape functions. It is seen from Eqs. 9 and 11 that $\underset{\sim}{H}$ is an mxm matrix which must be inverted to compute the element stiffness matrix. Furthermore, $\underset{\sim}{H}$ is fully populated. The single-element approach was rejected in view of the difficulty in finding 53 shape functions and the computational inefficiency associated with inversion of a 53 x 53 matrix.

A 90-degree segment of the near-field region was the first geometry for which an element development was attempted. The element shape, shown in Fig. 7, encompasses one quarter of the circular boundary and one edge of the near/far-field boundary. With 20 divisions allowed around the hole, the element possesses 6 inner nodes in addition to its 3 near/far-field boundary nodes.

The element can be subjected to rotation transformations and repeated assembly to model the entire near-field region. The kinematic stability requirement now leads to $m \geq 18 - 3 = 15$ for the minimum number of stress shape functions. However, since rotation transformations are contemplated, insensitivity to orientation enters as another practical requirement. This leads to a choice of m=16, as explained in Subsection 2.5. Tests of the 90-degree element highlighted a problem of inaccuracy in the integration for the $\underset{\sim}{H}$ matrix (Eq. 11), which was judged to have been caused by the presence of terms in sin(4θ) and cos(4θ) in the stress shape functions $\underset{\sim}{P}$.

The approach finally adopted divides the near-field region into eight 45-degree elements. Figure 8 illustrates the basic element, which possesses 4 nodes along its circular boundary. The element possesses 12 degrees of freedom, and therefore $m \geq 9$ is required. To preserve orientation insensitivity, m=10 is chosen, so that all sin(2θ) and cos(2θ) terms are included in $\underset{\sim}{P}$. The entire near-field region is assembled by first computing $\underset{\sim}{k}_1$ for the basic element and then subjecting $\underset{\sim}{k}_1$ to a series of transformations for the seven other elements shown in Fig. 9.

2.5 Final Formulation of Hybrid Hole Element

The choice of assumed-displacement interpolation functions for the hybrid element is dictated by the need to maintain inter-element compatibility along its edges. Since the hole element must be capable of coupling to a quadrilateral or a PCRK59 segment between each pair of nodes, linear interpolation must be used.

In general,

$$\begin{Bmatrix} u(s) \\ v(s) \end{Bmatrix} = \underset{\sim}{L}\,\underset{\sim}{q} = \begin{bmatrix} 0 & 0 & \cdots & 0 & 1-s/\ell & 0 & s/\ell & 0 & 0 & \cdots & 0 \\ 0 & 0 & \cdots & 0 & 0 & 1-s/\ell & 0 & s/\ell & 0 & \cdots & 0 \end{bmatrix} \begin{Bmatrix} q_1 \\ q_2 \\ \vdots \\ q_{12} \end{Bmatrix} \tag{13}$$

where q_1, q_2, $\ldots q_{12}$ are defined in Fig. 8, and where

$u(s)$ = Horizontal edge displacement.

$v(s)$ = Vertical edge displacement.

ℓ = Length of edge between a pair of nodes.

The arc coordinate s in Eq. 13 is a relative measure of distance along any edge, with the positive sense taken as counterclockwise for integration along ∂A. Note also that the nonzero terms in $\underset{\sim}{L}$ appear in different positions for different edges. For example,

$$\underset{\sim}{L}_{34} = \begin{bmatrix} 0 & 0 & 0 & 0 & 1-s/\ell & 0 & s/\ell & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1-s/\ell & 0 & s/\ell & 0 & 0 & 0 & 0 \end{bmatrix} \tag{14}$$

for the edge between nodes 3 and 4, while

$$\underset{\sim}{L}_{61} = \begin{bmatrix} s/\ell & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1-s/\ell & 0 \\ 0 & s/\ell & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1-s/\ell \end{bmatrix} \tag{15}$$

for the edge between nodes 6 and 1. In a similar manner,

$$\begin{aligned} s &= 0, \ell_{34} \quad \text{at nodes } 3,4 \\ s &= 0, \ell_{61} \quad \text{at nodes } 6,1 \end{aligned} \tag{16}$$

and so forth. These changes of definition do not cause difficulties in the boundary integration (Eq. 10) because the computation is carried out numerically, one edge at a time.

Assumed-stress shape functions are selected most conveniently by direct adaptation from classical elasticity solutions in polar coordinates. Let an Airy stress function $\Phi(r,\theta)$ be defined by:

$$\tau_{rr} = \frac{1}{r}\frac{\partial \Phi}{\partial r} + \frac{1}{r^2}\frac{\partial^2 \Phi}{\partial \theta^2} \qquad \tau_{\theta\theta} = \frac{\partial^2 \Phi}{\partial r^2} \qquad \tau_{r\theta} = \frac{1}{r^2}\frac{\partial \Phi}{\partial \theta} - \frac{1}{r}\frac{\partial^2 \Phi}{\partial r \partial \theta} \tag{17}$$

Then the equations of elasticity for homogeneous isotropic material reduce to:

$$\nabla^2 \nabla^2 \Phi(r,\theta) = 0 \tag{18}$$

where

$$\nabla^2 = \frac{\partial^2}{\partial r^2} + \frac{1}{r}\frac{\partial}{\partial r} + \frac{1}{r^2}\frac{\partial^2}{\partial \theta^2} \tag{19}$$

By assuming as an expansion for the stress function:

$$\Phi(r,\theta) = \phi^{(0)}(r) + \sum_{n=1}^{\infty} \phi^{(n)}(r)\left[\sin(n\theta) \text{ or } \cos(n\theta)\right] \tag{20}$$

it can be shown that Eq. 18 reduces to a set of equidimensional ordinary differential equations in $\phi^{(0)}(r)$, $\phi^{(1)}(r), \ldots$, for

which the following solutions are obtained [7]*:

$$\phi^{(0)}(r) = a_0 + b_0 \ln r + c_0 r^2 + d_0 r^2 \ln r \tag{21}$$

$$\phi^{(n)}(r) = a_n r^n + b_n r^{-n} + c_n r^{2+n} + d_n r^{2-n} \; ; \quad n \geq 2 \tag{22}$$

Stress distributions corresponding to Eqs. 21 and 22 are obtained by substituting back into Eqs. 17 [7]:

$$\begin{aligned} \tau_{rr}^{(0)} &= \frac{b_0}{r^2} + 2c_0 + d_0 (1 + 2\ln r) \\ \tau_{\theta\theta}^{(0)} &= -\frac{b_0}{r^2} + 2c_0 + d_0 (3 + 2\ln r) \\ \tau_{r\theta}^{(0)} &= 0 \end{aligned} \tag{23}$$

and

$$\begin{aligned} \tau_{rr}^{(n)} &= [\, a_n (n - n^2) r^{n-2} - b_n (n + n^2) r^{-n-2} + c_n (2 + n - n^2) r^n + \\ &\quad + d_n (2 - n - n^2) r^{-n}][\, \sin(n\theta) \text{ or } \cos(n\theta)] \\ \tau_{\theta\theta}^{(n)} &= [\, a_n (n^2 - n) r^{n-2} + b_n (n^2 + n) r^{-n-2} + c_n (2 + 3n + n^2) r^n + \\ &\quad + d_n (2 - 3n + n^2) r^{-n}][\sin(n\theta) \text{ or } \cos(n\theta)] \\ \tau_{r\theta}^{(n)} &= [\, a_n (n^2 - n) r^{n-2} - b_n (n^2 + n) r^{-n-2} + c_n (n^2 + n) r^n + \\ &\quad - d_n (n^2 - n) r^{-n}][-\cos(n\theta) \text{ or } \sin(n\theta)] \end{aligned} \tag{24}$$

*The solution for $\phi^{(1)}(r)$ is of no interest in the present work. Terms in $\sin\theta$ and $\cos\theta$ correspond to problems in which a discontinuity has been introduced, e.g., re-welding of an annulus after a segment has been cut out.

The undetermined coefficients b_o, c_o, ... , d_n in Eqs. 23 and 24 play the role of assumed-stress degrees of freedom, which are eliminated when the element stiffness matrix is formed [1]. The remaining $r\theta$-dependent portions of the terms are placed in the shape function matrix $\underset{\sim}{P}$.

Shape functions have been chosen from Eqs. 23 and 24 as follows. For the θ-independent terms, logarithmic behavior is not particularly desirable; only the b_o and c_o terms have been retained. This choice is admittedly arbitrary. To complete $\underset{\sim}{P}$, enough θ-dependent terms are chosen from 2θ, 4θ, 8θ,... behavior to satisfy kinematic stability and orientation insensitivity. The latter requirement demands, e.g., that all 2θ-dependent terms be retained if any are required for stability. It is apparent from Eqs. 24 that there are eight $n\theta$-dependent terms for each n when $\sin(n\theta)$ and $\cos(n\theta)$ terms are included. Thus, selection of the 2θ-dependent terms provides a total of 10 shape functions, just enough to satisfy the requirements for the 45-degree element (The 90-degree element requires addition of the 4θ-dependent terms as well, making a total of 18 shape functions.). The shape function matrix, as finally adopted for the 45-degree element, is given by:

$$\underset{\sim}{P}(r,\theta) = \left\{ \begin{array}{ccccc} -2\cos 2\theta & -6\cos 2\theta/r^4 & 0 & -4\cos 2\theta/r^2 & -2\sin 2\theta \\ 2\cos 2\theta & 6\cos 2\theta/r^4 & 12r^2\cos 2\theta & 0 & 2\sin 2\theta \\ 2\sin 2\theta & -6\sin 2\theta/r^4 & 6r^2\sin 2\theta & -2\sin 2\theta/r^2 & -2\cos 2\theta \end{array} \right.$$

$$\left. \begin{array}{ccccc} -6\sin 2\theta/r^4 & 0 & -4\sin 2\theta/r^2 & 1/r^2 & 2 \\ 6\sin 2\theta/r^4 & 12r^2\sin 2\theta & 0 & -1/r^2 & 2 \\ 6\cos 2\theta/r^4 & -6r^2\cos 2\theta & 2\cos 2\theta/r^2 & 0 & 0 \end{array} \right\} \quad (25)$$

The numerical integration required for computation of the $\underset{\sim}{H}$ matrix (Eq. 11) is accomplished most conveniently in polar coordinates. However, computation of $\underset{\sim}{G}$ (Eq.10) actually requires the cartesian surface tractions which correspond to the assumed-displacement field:

$$\underset{\sim}{T} = \{T_x \ T_y\} = \underset{\sim}{N}\{\sigma_{xx} \ \sigma_{yy} \ \sigma_{xy}\} = \underset{\sim}{N}\underset{\sim}{P}\underset{\sim}{\beta} \tag{26}$$

where $\underset{\sim}{\beta}$ represents the vector of stress unknowns [1]. Hence, the following modification of Eq. 10 is required. Since

$$\underset{\sim}{\tau} = \{\tau_{rr} \ \tau_{\theta\theta} \ \tau_{r\theta}\} = \underset{\sim}{P}\underset{\sim}{\beta} \tag{27}$$

in the present case, a Mohr circle transformation must be introduced:

$$\underset{\sim}{\sigma} = \underset{\sim}{M}\underset{\sim}{\tau} = \underset{\sim}{M}\underset{\sim}{P}\underset{\sim}{\beta} \tag{28}$$

and Eq. 10 is replaced by:

$$\underset{\sim}{G} = t\int_{\partial A} (\underset{\sim}{N}\underset{\sim}{M}\underset{\sim}{P})^T \underset{\sim}{L} \, dA \tag{29}$$

where

$$\underset{\sim}{M} = \begin{bmatrix} \cos^2\theta & \sin^2\theta & -\sin 2\theta \\ \sin^2\theta & \cos^2\theta & \sin 2\theta \\ \frac{1}{2}\sin 2\theta & -\frac{1}{2}\sin 2\theta & \cos 2\theta \end{bmatrix} \tag{30}$$

In Eq. 30, θ is the angular coordinate of the current integration station on ∂A. Thus, the price paid for convenience in choosing

$\underset{\sim}{P}(r,\theta)$ is the slight amount of additional computation required to form $\underset{\sim}{M}$ at each integration station, as the element boundary ∂A is swept in Eq. 29.

2.6 Transformation-Assembly Procedure

A complete set of stiffnesses for the near-field region can be obtained from a single element computation by means of a transformation-assembly sequence. Element number 1 in Figs. 8 and 9 is always assumed to be oriented with edge 61 along the positive x-axis. Eqs. 11 and 29 are integrated with Gauss-Lagrange quadrature (GLQ) formulas [8] to compute $\underset{\sim}{k}_1$, which is then assembled into a near-field substructure in accordance with the global node numbering scheme in Fig. 9.

It is apparent that the stiffnesses for elements 3, 5 and 7 can be obtained by rotation transformation of $\underset{\sim}{k}_1$. For example, element 3 is oriented at +90 degrees from element 1. Hence, the transformation:

$$\{q_1 \; q_2 \cdots q_{12}\}_L = \underset{\sim}{R} \{q_1 \; q_2 \cdots q_{12}\}_G \tag{31}$$

relates the locally oriented displacements $\underset{\sim}{q}_L$ to the globally oriented displacements $\underset{\sim}{q}_G$, where:

$$
\underset{\sim}{R} = \begin{bmatrix} \cos\theta & \sin\theta & & & & \\ -\sin\theta & \cos\theta & & & & \\ & & \cos\theta & \sin\theta & & \\ & & -\sin\theta & \cos\theta & & \\ & & & & \ddots & & \\ & \text{(Super-diagonal)} & & & & \cos\theta & \sin\theta \\ & & & & & -\sin\theta & \cos\theta \end{bmatrix}_{(12\times 12)} \tag{32}
$$

An example for one pair of nodal displacements is illustrated in Fig. 9. Since strain energy is independent of any particular coordinate system, a global stiffness matrix for element 3 may be computed as follows:

$$
\text{Strain Energy} = \tfrac{1}{2}\, \underset{\sim}{q}_L^T\, \underset{\sim}{k}_1\, \underset{\sim}{q}_L = \tfrac{1}{2}\, \underset{\sim}{q}_G^T\, \underset{\sim}{k}_3\, \underset{\sim}{q}_G \tag{33}
$$

Introducing Eq. 31 for $\underset{\sim}{q}_L$ leads to

$$
\underset{\sim}{q}_G^T\, \underset{\sim}{R}^T\, \underset{\sim}{k}_1\, \underset{\sim}{R}\, \underset{\sim}{q}_G = \underset{\sim}{q}_G^T\, \underset{\sim}{k}_3\, \underset{\sim}{q}_G \tag{34}
$$

for arbitrary values of $\underset{\sim}{q}_G$. Therefore:

$$
\underset{\sim}{k}_3 = \underset{\sim}{R}^T\, \underset{\sim}{k}_1\, \underset{\sim}{R} \quad ; \quad \theta = \pi/2 \tag{35}
$$

and in a similar manner,

$$
\underset{\sim}{k}_5,\ \underset{\sim}{k}_7 = \underset{\sim}{R}^T\, \underset{\sim}{k}_1\, \underset{\sim}{R} \quad ; \quad \theta = \pi,\ 3\pi/2 \tag{36}
$$

The sequence of rotation transformations given by Eqs. 35 and 36 assembly according to Fig. 9 is followed.

A stiffness matrix for element 2 can be obtained from $\underset{\sim}{k}_1$ by the reflection transformation illustrated in Fig. 10. If element 2 is considered relative to a reflected xy axis system, then its stiffness is identical to $\underset{\sim}{k}_1$ with the displacements oriented as indicated by the broken arrows. However, stiffness $\underset{\sim}{k}_2$ is wanted again with respect to $\underset{\sim}{q}_G$. It is evident from Fig. 10 that for this case:

$$\underset{\sim}{q}_L = \underset{\sim}{\Lambda}\,\underset{\sim}{q}_G \tag{37}$$

where

$$\underset{\sim}{\Lambda} = \begin{bmatrix} 1 & & & & & \\ & -1 & & & & \\ & & 1 & & & \\ & & & -1 & & \\ & & & & \ddots & \\ (Diagonal) & & & & & 1 \\ & & & & & & -1 \end{bmatrix}_{(12 \times 12)} = \underset{\sim}{\Lambda}^T \tag{38}$$

Applying the strain energy argument again then gives immediately:

$$\underset{\sim}{k}_2 = \underset{\sim}{\Lambda}\,\underset{\sim}{k}_1\,\underset{\sim}{\Lambda} \tag{39}$$

After $\underset{\sim}{k}_2$ has been assembled, the procedure is completed with rotation transformations for:

$$\underset{\sim}{k}_4, \underset{\sim}{k}_6, \underset{\sim}{k}_8 = \underset{\sim}{R}^T \underset{\sim}{k}_2 \underset{\sim}{R} \quad ; \quad \theta = \pi/2,\ \pi,\ 3\pi/2 \tag{40}$$

where $\underset{\sim}{R}$ is given by Eq. 32. The transformation-assembly procedure has been programmed internally, and the final result may be thought of as a single element with 16 outer boundary nodes, 24 inner boundary nodes, and 64 total degrees of freedom. Figure 11 illustrates the numbering convention for the assembly, which will be referred to as HOLEL in subsequent discussion.

2.7 Performance Tests

Several performance tests were conducted, in addition to routine checking of inversion accuracy in the $\underset{\sim}{H}$ matrix, in order to assess the capabilities of HOLEL. Although its primary function is to serve as a link between a few inner rings of quadrilaterals and a coarse far-field mesh, there is some interest in determining the solution accuracy for structures in which the inner boundary of HOLEL is stress-free. Errors are to be expected for this situation, since the assumed surface tractions $\underset{\sim}{N}\,\underset{\sim}{M}\,\underset{\sim}{P}\,\underset{\sim}{\beta}$ generally do not vanish when $\underset{\sim}{P}(r,\theta)$ is computed for points lying on the element boundary ∂A. Errors of this type usually appear as overshoots in the computed stresses, given by [1]:

$$\underset{\sim}{\tau}(r,\theta) = \underset{\sim}{P}(r,\theta)\,\underset{\sim}{H}^{-1}\,\underset{\sim}{G}\,\underset{\sim}{q}_L = \underset{\sim}{B}(r,\theta)\,\underset{\sim}{q}_L \qquad (41)$$

where r,θ represent any point in the element domain A or on the boundary ∂A. Matrices $B_1(r,\theta)$ were computed during the formation of $\underset{\sim}{k}_1$ for several points along a ray at $\theta = 22.5$ degrees in element 1 (Fig. 9) for the purpose of the test.

Figs. 13, 14 and 15 illustrate the results of a test series in which the dimensions of HOLEL are (see Fig. 12):

Hole Diameter D_o = 2 inches

Edge Dimension W = 8 inches

The value W/D_o = 4 was chosen on the basis of a classical elasticity solution which indicates that far-field conditions are achieved at r/R_o = 4 for an open hole in an infinite plate [9]. The finite-element results computed from Eq. 41 are compared with the classical solution at θ = 22.5 degrees [7].

The effect of increasing the number of GLQ integration stations is studied in Fig. 13, which plots $\tau_{\theta\theta}$. This stress component is the least sensitive to overshoot error because it does not enter into the stress-free condition on the inner circular portion of the boundary. The results show that 3 GLQ stations are too few, while acceptable accuracy is obtained with 4 or more stations. The programming of HOLEL was subsequently fixed at 5 GLQ stations based on these results. The meaning of "5 stations" is actually:

5 stations x 6 edges to compute G on ∂A	= 30
5 x 5 stations to compute H in A	= 25
Total GLQ integration stations	= 55

This amount of computing is comparable to the amount required for the PCRK59 element.

Figures 14 and 15 illustrate the behavior of τ_{rr} and $\tau_{r\theta}$, respectively. Finite-element results are shown only for 5 GLQ stations. These results demonstrate surprisingly that HOLEL is able to achieve the stress-free condition. However, some inaccuracy

can be observed at $r/R_o \cong 1.7$. The important conclusion to be drawn from this test series is that HOLEL can be used without interior elements to model a multi-fastener-hole structure for which, e.g., local stress distributions or K_I and K_{II} solutions are sought only at one fastener hole, or at some other location.

The second test series studied the performance of HOLEL as a function of the parameter W/D_o, using the test problem illustrated in Fig. 12. Results for $\tau_{\theta\theta}$ are plotted in Fig. 16. Some performance degradation is observed for $W/D_o = 3$, as the near-field stress gradients begin to appear in the far-field elements, which are incapable of following this behavior with complete accuracy. Considerable degradation is also seen for $W/D_o = 8$, a result which might be attributed to either the absence of far-field terms in $\underset{\sim}{P}(r,\theta)$, or errors in computing $\underset{\sim}{G}$ and $\underset{\sim}{H}$ caused by shape distortion, or both. One may conclude from these results that limitations must be placed on the edge distance and centerline spacing in HOLEL models of multi-fastener structures.

The objective of the third test series was to study the accuracy of HOLEL for the case of combined interference-fit or bearing loads and panel tension. Two tests were conducted, in which additional loads were applied at the fastener hole boundary of the model shown in Fig. 12. In Fig. 17, results for τ_{rr} are plotted for the case of 1 ksi panel tension combined with 1 ksi uniform pressure to represent an interference fit. A curve faired through the computed stresses extrapolates to -1.07 ksi at the fastener hole boundary, i.e., an error of about 7 percent. In

Fig. 18, results for τ_{rr} are plotted for 1 ksi panel tension combined with a bearing pressure distribution given by:

$$p(\theta) = \frac{2P}{\pi R_0} \sin\theta \quad ; \qquad P = 8{,}000 \text{ lb.} \; , \quad 0 \leq \theta \leq \pi \tag{42}$$

Eq. 42 gives

$$\tau_{rr} = - p(\theta) = -1.95 \text{ ksi} \tag{43}$$

at θ = 22.5 degrees. In this case, a curve faired from 11 computed stresses extrapolates to the exact value at the fastener hole boundary. It is apparent from these results that HOLEL is accurate for both interference-fit and bearing loads. In the former case, the results could have been improved by computing τ_{rr} at 11 locations instead of the 5 which had been used. Far-field element stresses were also checked for this test series. In the case of interference-fit, the computed values for cartesian stress in the far-field elements were σ_{yy} = 1 ksi and $\sigma_{xx} = \sigma_{yy} \cong 0$ (to the roundoff level of the computer), with very little disturbance. In the case of bearing, it must be recognized that the 8,000-lb. bearing load adds to the 16,000-lb. panel tension load (1 ksi x 16-inch edge) for the far-field elements below the fastener hole. The computed stresses were in fact σ_{yy} = 1 ksi for the upper elements, σ_{yy} = 1.5 ksi for the lower elements, and $\sigma_{xx} = \sigma_{xy} \cong 0$, with load-transfer effects appearing in the far-field elements adjacent to HOLEL.

In the fourth test series, four rings of quadrilateral and PCRK59 elements were added to the structure model with the outer ring coupled to the circular boundary of HOLEL. Two test cases were analyzed. In the first case, the ring consisted entirely of quadrilaterals, and a cosine bearing pressure distribution (Eq. 42) was applied at the free boundary of the innermost ring. Figure 19 plots the results for τ_{rr} computed at the centroids of one "slice" of four quadrilaterals, along a ray at θ = 97.5 degrees. The faired-curve extrapolation of these four data points agrees well with

$$\tau_{rr} = -p(97.5^{\circ}) = -\frac{2P}{\pi R_0} \sin(97.5^{\circ}) \cong -25.2 \text{ ksi} \tag{44}$$

In the second case, one or two groups of four quadrilaterals were replaced by PCRK59 elements to simulate a fastener hole with a crack at $\theta = 0$ or two equal-length cracks at $\theta = 0, \pi$. Several cases were analyzed covering the parameter ranges:

Hole diameter/HOLEL Edge: $0.02 < D_I/W < 0.05$

HOLEL edge/HOLEL diameter: $W/D_O = 4$

Crack length/hole radius: $0.25 < a/R_I < 1.0$

Table 1 compares the computed K_I values with independent classical solutions [10, 11]. The results demonstrate that the finite-element model is capable of providing reasonably accurate K_I solutions for quite extreme geometries (small fastener hole in a large panel).

The final test series consisted of a number of demonstration runs of the complete PANEL program, in which K_I and K_{II} solutions

were computed for another range of W/D_o values. These results are presented in Section 5, following the documentation of the HOLEL procedure and the PANEL program.

Section 3

HOLEL PROCEDURE

Due to the complexity of the model, HOLEL has been programmed as several interrelated subroutines. Subroutine HOLEL provides overall control of the stiffness generation-assembly-transformation procedure, calling upon subroutines QHOLEL, GMTRX, HMTRX, LMTRX, MNMTRX, PMTRX and TRIG to execute specific computational tasks. In addition, HOLEL uses some of the FEABL-2 software [6] for the assembly process.

3.1 Structure Model and Input/Output Conventions

A square near-field region with an inner circular boundary of radius R_o is modelled. The numbering conventions have been indicated in Fig. 11. Subroutine HOLEL is invoked by:

CALL HOLEL(COORD,THK,S,RI,RSS,ISS,B)

where the arguments are dimensioned and defined as follows:

COORD(12) - Vector of cartesian coordinates of the element corners in order $X_1,Y_1,Z_1,X_2,\ldots,Z_4$. (In the present version, the element is assumed to lie in the XY plane, and the Z coordinates need not be given any values.)

THK - Scalar value of element thickness.

S(3,3) - Array of elastic compliance constants for homogeneous isotropic material, i.e.:

$$\underset{\sim}{S} = \frac{1}{E}\begin{bmatrix} 1 & -\nu & 0 \\ -\nu & 1 & 0 \\ 0 & 0 & 2(1+\nu) \end{bmatrix} \tag{45}$$

RI - Scalar value of inner boundary radius R_o.

RSS(2097), ISS(2097) - Floating-point and fixed-point names of a FEABL-2 DATA vector. These arguments must be equivalenced, as well as dimensioned in the user's MAIN program.

B(6,3,13) - A collection of $\underset{\sim}{B}$ matrices for stress analysis (see Subsection 3.2).

HOLEL returns the assembled subregion stiffnesses as a variable-bandwidth-stored matrix in the DATA vector RSS,ISS. The results may be read into another storage area with the algorithm given in Appendix A, or may be assembled directly into another DATA vector with FEABL-2 subroutine ASMSUB (see Subsection 2.2). Procedure HOLEL requires no input data cards.

3.2 Required Subprograms and Other Features

Procedure HOLEL requires the following additional software for execution:

1. ASRL FEABL-2 subroutines ASMLTV, ORK and SETUP.
2. IBM Scientific Subroutine Package subroutines MFSD and SINV.

No external disk or tape files are required.

3.3 Model Generation and Program Flow

Figure 20 summarizes the program flow in Procedure HOLEL. At entry, whatever control parameters are present in the FEABL-2 labelled COMMON areas SIZE, BEGIN and END are saved in temporary storage. Subroutine SETUP is called to establish address control for the RSS,ISS vector for a structure model with 64 total degrees of freedom, consisting of the eight 12-degree-of-freedom elements in Fig. 9. Element interconnections are then generated internally, in accordance with the numbering conventions in Fig. 8 and Fig. 11, and subroutine ORK is called to compute the band margin and complete address control for the assembled stiffness matrix.

Subroutine QHOLEL is now called to compute $\underset{\sim}{k}_1$, the stiffness matrix for element 1. This subroutine calls in turn subroutines HMTRX and GMTRX to compute $\underset{\sim}{H}^{-1}$ and $\underset{\sim}{G}$, forms the stiffnesses $\underset{\sim}{G}^T\underset{\sim}{H}^{-1}\underset{\sim}{G}$. Subroutines HMTRX and GMTRX loop over the GLQ integration stations in the element domain A and on ∂A respectively, calling upon subroutines TRIG, PMTRX, LMTRX and MNMTRX to compute $\underset{\sim}{P}(r,\theta)$, $\underset{\sim}{L}(A)$ and $\underset{\sim}{M}^T\underset{\sim}{N}^T$ at each station, and accumulating the product sums (including GLQ weighting factors) for $\underset{\sim}{P}^T\underset{\sim}{S}\underset{\sim}{P}$ and $(\underset{\sim}{N}\underset{\sim}{M}\underset{\sim}{P})^T\underset{\sim}{L}$. Subroutine HMTRX calls SINV (which calls MFSD) to invert $\underset{\sim}{H}$.

After $\underset{\sim}{k}_1$ has been computed, subroutine QHOLEL forms a 6x3x13 matrix for stress analysis, consisting of six matrices $\underset{\sim}{B}(r,\theta)$ as given by Eq. 41, and stored in accordance with the following conventions:

$$B(I,J,K) = \left[\underset{\sim}{B}_i(r_i,\theta_i) \;\vdots\; \begin{matrix}\theta_i \\ r_i \\ 0\end{matrix} \right] \qquad (46)$$

where subscript I identifies the specific $\underset{\sim}{B}$ matrix, and where

$$
\begin{aligned}
\theta_i &= \pi/8 \quad ; \quad i = 1, 2, \cdots, 6 \\
r_1 &= R_o \\
r_i &= r_{i-1} + 0.2\,(\text{distance from } R_o \text{ to outer} \\
&\qquad \text{boundary along } \theta_i) \; ; \; i = 2, 3, \cdots, 6
\end{aligned}
\tag{47}
$$

Thus, the $\underset{\sim}{B}$ matrices refer to six locations which divide the ray $\theta = \pi/8$ into five equal segments, with $\underset{\sim}{B}_1$ located on the inner circular boundary and $\underset{\sim}{B}_6$ located on the near/far-field boundary. Polar stresses for these locations can be computed from:

$$
\{\tau_{rr} \;\; \tau_{\theta\theta} \;\; \tau_{r\theta}\}_i = STRESS(J)_I = \sum_{K=1}^{12} B(I,J,K) * Q(K) \tag{48}
$$

where

$$
Q(K) = \underset{\sim}{q}_L \tag{49}
$$

An extra column (K=13) is appended to each $\underset{\sim}{B}$ matrix for storage of the polar coordinates of its location.

At this point, subroutine HOLEL regains control and executes the transformation-assembly process outlined in Subsection 2.6. In the program, $\underset{\sim}{k}_2$ is formed first, and the assembly sequence is programmed as:

Assembly of $\underset{\sim}{k}_1$ and $\underset{\sim}{k}_2$

Rotation through $\pi/_2$ to $\underset{\sim}{k}_3$ and $\underset{\sim}{k}_4$

Assembly of $\underset{\sim}{k}_3$ and $\underset{\sim}{k}_4$

Rotation through π to $\underset{\sim}{k}_5$ and $\underset{\sim}{k}_6$

.
.
.

Assembly of $\underset{\sim}{k}_7$ and $\underset{\sim}{k}_8$

Finally, the HOLEL address control parameters are transferred to the FEABL-2 labelled COMMON areas SIZESS and BEGSS, and the user's control parameters are returned to areas SIZE, BEGIN and END. Control is now returned to the user's calling program, with HOLEL ready for assembly to the user's structure model by means of FEABL-2 subroutine ASMSUB.

3.4 Procedure Status

All of the HOLEL subroutines, together with copies of IBM subroutines MFSD and SINV, are maintained as a unit in the form of individually sequenced, 029-punched FORTRAN-IV source decks. A listing of Procedure HOLEL appears in Appendix B.

Section 4

PANEL PROGRAM

The PANEL program is an executive program which controls the assembly of several substructures into a model of a skin tension panel with a single open fastener hole. The PANEL program uses procedure HOLEL and several additional special-purpose procedures to generate the required substructures. Details of these additional procedures are discussed in Subsections 4.2 and 4.3.

4.1 Structure Model and Input Conventions

Figure 21 illustrates the structure which the PANEL program models: a rectangular panel loaded by uniform tension on its horizontal edges. The panel contains a single fastener hole centered vertically. The fastener hole may be either centered or offset horizontally. The vertical edges of the panel may be stiffened symmetrically with integral stiffeners, if desired. One or two cracks may be placed to emanate radially from the fastener hole. If two cracks are specified, they will be located 180 degrees apart, and they may be of equal or unequal length. The angular position, θ, to the first crack may be varied in 15-degree increments from 0 to 345 degrees if there is only one crack, or from 0 to 165 degrees if there are two cracks. A second version of the PANEL program is available for a structure in which only the left edge may be stiffened. The second version

is otherwise similar to the first version, and will not be discussed separately.

Four input data cards are required for the PANEL program. The input data describe the specific panel geometry and define several parameters which control the type of solution executed and the amount of information printed. Figure 22 illustrates the correct format for the input cards:

1. Panel parameters

   WIDTH = Total width of the panel, W_p.

   LENGTH = Total length of the panel.

   THK = Panel thickness.

   STFFCT = Stiffener factor.

   PRESS = Tension loading, σ_{yy}.

   RI = Radius of the fastener hole.

   IOFFST = Offset indicator.

2. Crack parameters

   A(1) = Length of first crack.

   A(2) = Length of second crack.

   IPOS(1) = Crack initial position number

   IPOS(2) = Crack final position number.

3. Material properties

   E = Young's modulus.

   ν = Poisson's ratio.

4. Print control parameters

   KT1 = Control for optional FEABL-2 output.

   KT2 = Control for optional FARFLD output.

   KT3 = Control for optional RING output.

The stiffener factor is defined in terms of the panel dimensions (Fig. 21):

$$STFFCT = W_r t_r / W_p t_p \tag{50}$$

The value STFFCT = 0 corresponds to an unstiffened panel. Negative values may be used to analyze panels with edges thinner than the primary structure. The offset indicator is used to control the type of solution desired. The following options are available:

IOFFST = 0 - Centered fastener hole only.

IOFFST = 1 - Fastener hole moves right.

IOFFST = -1 - Fastener hole moves left.

For the latter two options, solutions are executed automatically beginning with the hole on center and ending with the hole as close to the edge of the panel as permitted by the finite-element model. If STFFCT ≠ 0, motion of the hole is further restricted to avoid overlapping with a stiffener.

A value of 0 should be assigned to crack length A(2) if a structure with only one crack is to be analyzed. Crack sizes up to 1.27 (RI) are permitted. The crack position numbers control the angular location of the first crack as follows:

$$\begin{aligned} \theta_{initial} &= (IPOS(1) - 1)\,\Delta\theta \\ \theta_{final} &= (IPOS(2) - 1)\,\Delta\theta \\ \Delta\theta &= \pi/12 \text{ rad.} = 15 \text{ deg.} \end{aligned} \tag{51}$$

Permissible limits for the position numbers are:

$$1 \leq IPOS(1) \leq n_{max} \qquad IPOS(1) \leq IPOS(2) \leq n_{max} \tag{52}$$

where

$$\begin{aligned} n_{max} &= 24 \quad \text{if} \quad A(2) = 0 \\ n_{max} &= 12 \quad \text{if} \quad A(2) \neq 0 \end{aligned} \tag{53}$$

Solutions are executed automatically beginning with the first crack in position 1 and ending with the first crack in position 2. A single solution is executed if IPOS(2) = IPOS(1). A case matrix is executed if ICFFST ≠ 0 and IPOS(2) > IPOS(1).

Most of the routine information printed by the PANEL program is of no interest when production runs are executed. This information may be deleted from the output by assigning each of the control parameters KT1, KT2, KT3 values equal to the FORTRAN unit number for the line printer at the user's computing facility.* Any other value permits full output by the associated software. Full output is recommended for initial testing of the program at a new facility. Output from FARFLD and RING should be allowed whenever a new range of panel dimensions is tried.

4.2 Required Subprograms and Other Features

The PANEL program requires the following additional software for execution:

*The same value used in a print instruction, e.g., WRITE (6,1000) A,B,C. The FORTRAN unit number in this case is 6.

1. ASRL FEABL-2 subroutines ASMLTV, ASMSUB, BCON, FACT, ORK, QBACK, SETUP, SIMULQ, STACON and XTRACT.
2. IBM Scientific Subroutine Package subroutines MFSD and SINV.
3. ASRL procedures FARFLD, HOLEL and RING, with included subroutines.
4. ASRL elements PCRK59 and QUAD4.

No external disk or tape files are required. The program must be able to communicate with the user's facility card reader and line printer, by means of two instructions near the beginning of the program:

KR = Card reader FORTRAN unit number.

KW = Line printer FORTRAN unit number.

The program is supplied with IBM-standard values KR=5 and KW=6.

4.3 Model Generation and Program Flow

The panel structure model is created via several levels of substructuring to minimize repeated processing of unwanted degrees of freedom. Figure 23 illustrates the general hierarchy of the finite-element model. Procedures FARFLD and RING generate the two major components which are finally assembled to form the complete structure.

Procedure FARFLD creates the far-field region, consisting of upper and lower rectangular substructures generated by procedure LUG,* and a center-zone substructure generated by

*Different from the attachment lug procedure reported in Ref. 1.

procedure CZONE. Procedure LUG assembles a regular mesh of QUAD4 elements and eliminates all except the upper-edge and lower-edge nodes. Procedure CZONE assembles HOLEL together with right and left portions modelled with QUAD4 elements, and eliminates all but the upper- and lower-edge nodes and the inner circular boundary nodes. Procedure FARFLD assembles the LUG and CZONE components and executes additional Gauss elimination to produce one of two major substructures:

1. "Panel-and-Hole", in which only the nodes along the top and bottom edges of the panel and the nodes along the inner circular boundary remain as boundary nodes in the statically condensed structure. Displacement restraints are applied at the bottom edge and nodal forces are applied at the top edge to represent uniform tension.
2. "The Cheshire Cat": a panel-and-hole with top and bottom edges eliminated by static condensation (nothing remains but the smile).

Only the Cheshire Cat option is used by the PANEL program. Numbering conventions for the FARFLD components are illustrated in Figs. 24, 25 and 26.

Procedure RING creates an inner cracked ring for assembly inside the Cheshire Cat. The ring consists of two 150-degree arcs and two 30-degree segments which contain QUAD4 and PCRK59 elements. The arcs are assembled by procedure ARC4, which also

removes all interior nodes by static condensation. Procedure RING assigns the angular locations of the components to obtain the required crack positions, and determines where each PCRK59 element is to be placed in the 30-degree segments according to the crack sizes A(1), A(2). Finally, procedure RING assembles and statically condenses the cracked ring to produce one of two major substructures:

1. Ring with all but inner and outer circular boundary nodes eliminated.
2. Ring with all but outer circular boundary nodes eliminated.

Retention of the inner boundary nodes is useful for cases in which bearing or interference-fit loads are to be applied at the fastener hole. The outer boundary nodes must be retained for coupling with the Cheshire Cat. The PANEL program uses only the second option. Numbering conventions for the RING components are illustrated in Figs. 27 through 30. Procedure RING always generates a ring with an outer/inner diameter ratio of 2.52 to maintain shape conformity for the QUAD4 elements.

Figure 31 illustrates the executive flow in the PANEL program. After the problem input data have been read and printed and some parameters have been calculated for control of the various procedures, two major loops appear. The outer loop begins with the creation of a Cheshire Cat, a step which must be repeated each time the fastener hole is offset to a new position. This is followed by address control and interconnection generation for

the final structure, which will be assembled from the Cheshire Cat (super-element 1) and the cracked ring (super-element 2). The interconnection algorithm is actually completed inside the inner loop to allow the band-margin computations (Subroutine ORK) to erase the $\underset{\sim}{K}$ and $\underset{\sim}{q}$ data from the previous pass. The remainder of the inner loop consists of assembly of the Cheshire Cat, generation and assembly of the cracked ring, global solution (subroutines FACT and SIMULQ), a back-substitution to obtain $\underset{\sim}{q}_L$ for the cracked ring substructure and finally, extraction of the PCRK59 displacements from $\underset{\sim}{q}_L$ and computation of K_I and K_{II}. The ends of the inner and outer loops are governed by incrementation of the crack angle and hole offset, respectively, and logical checks to determine whether the prescribed ranges of these parameters have been swept.

4.4 Output Conventions and Error Messages

Figure 32 illustrates a sample output from the PANEL program (version 2, left side stiffened). The output from Version 1 is similar. The heading identifies the program version and repeats the user's input data. Below the material properties data appears a table of K_I and K_{II} solutions for one or two cracks, together with their angular positions. The K_I and K_{II} values are NASA/ASTM standard stress intensities in units of psi $\sqrt{in.}$, assuming that the input data was specified in corresponding engineering units of psi for loading and Young's modulus and inches for dimensions. The sample output is an example of the production information obtained by exercising the three options for deletion of debugging output.

Besides the FEABL-2 software error messages [6], the following diagnostics may result from the PANEL programs. In procedure RING, the length specified for each crack is checked to insure that the crack tip does not extend beyond the average radius of the outermost ring of quadrilaterals, i.e.:

$$R_I + A(J) \leq \frac{1}{2}\left(R_4 + 2.52 R_I\right) ; \quad J = 1, 2 \tag{54}$$

where R_4 is the inner radius of the outermost ring. If Eq. 54 is violated, a message is printed and program execution is terminated. This condition is somewhat conservative, since it restricts the crack tip to a position mid-way in the PCRK59 element. Eq. 54 may be replaced by:

$$R_I + A(J) \leq 0.3 R_4 + 0.7 \times 2.52 R_I \tag{55}$$

to permit the crack tip to approach somewhat closer to the outer boundary. In procedure CZONE, a check is made to insure that the fastener hole offset is within allowable limits. This is done by monitoring the node number of the fictitious center reference node (see Fig. 25). Excessive offset causes an error message and program termination.

4.5 Program Status

Both versions of the PANEL program have been exercised successfully for all analysis options, and are maintained as

sequenced, 029-punched FORTRAN-IV source decks. Either version requires 260 to 300 KBYTES ($65,000_{10}$ to $75,000_{10}$ words, or $176,750_8$ to $222,370_8$ words) of core memory and approximately 0.8 to 1.0 CPU minute, depending upon the ranges of the hole offset and crack angle parameters. Storage and time statistics are based on runs made on an IBM S-370/168 machine, using the IBM FORTRAN-G1 and FORTRAN-H compilers. Version 1 (symmetrical edge stiffeners) is listed in Appendix C. Version 2 (left edge stiffener only) is listed in Appendix D.

Section 5

DEMONSTRATION EXAMPLES

A number of example analyses have been run to verify the PANEL program code and simultaneously to explore the accuracy of the analysis for a range of panel and crack dimensions wider than considered in the HOLEL performance tests. The solutions given here focus mainly on panels with centered fastener holes having cracks at $\theta=0,\pi$ because there exist no independent solutions with which to compare other configurations.

Figure 33 illustrates the coarsest model possible to generate from the PANEL program: lugs consisting of two elements each and a center-zone composed only of HOLEL. Also, the dimensions chosen are such that the fastener hole is no longer small compared to the panel, and since the outer/inner diameter ratio of the cracked ring is 2.52, the HOLEL shape parameter in this case is:

$$W/D_o = 4/2.52 \cong 1.59$$

a point well outside the range studied in the HOLEL performance tests. The butterfly plots for K_I and K_{II} shown in Fig. 33 illustrate an error effect caused by proximity of the displacement boundary conditions to the region of interest. In the present case, the restraints are only two elements away from the "action" (a lug QUAD4 and the HOLEL) when either crack lies below the

horizontal. The error is most pronounced for K_{II} at $\theta = 225$, 315 degrees. The error would be more pronounced as $\theta \rightarrow 270$ degrees, except that the solution tends rapidly to zero in this region. A similar but less-pronounced effect can be observed in the plot for K_I. Comparison of K_I for $\theta = 0$, 180 degrees with an independent solution [10, 11] provides a more welcome result. Indeed, it is quite surprising that this very crude finite-element model is able to faithfully reproduce the classical solution. The general conclusion to be drawn from this test is that only the upper half of the butterfly plot may be trusted when running models with very few elements in the FARFLD substructure.

Figures 34 and 35 illustrate two runs in which the panel and crack dimensions have been varied to explore the effect of W/D_O. Also, the aspect ratios of the elements in the LUG substructure were changed to allow more elements between the restrained bottom edge and the center-zone. The latter modification has eliminated the restraint error effect. Table 2 summarizes the comparisons of results from Figs. 33 through 35 with the independent solution, showing that reasonable accuracy has been achieved over:

$$1.59 \leq W/D_O \leq 6.35$$

The greater error for the middle case remains an unexplained anomaly.

Figure 36 illustrates the K_I and K_{II} solutions obtained for a panel of the same dimensions, hole radius, etc. shown in Fig. 35, but with the hole offset to the maximum amount permitted in

the model, a distance of 1.5 inches. Increases of the order of 5 to 10 percent are observed for K_I, with the greatest increase at crack tips which are located nearest to the edge of the panel.

Figures 37 through 39 present results for some additional studies of the basic panel shown in Fig. 35 to illustrate the stiffening options. A stiffener factor of 0.5 was used for these analyses, i.e., the added cross section area of each stiffener was 50 percent of the panel cross section area. Figure 37 plots K_I and K_{II} solutions for a symmetrically stiffened panel. Results are shown for the fastener hole centered and offset by 1 inch. There is very little difference between the two solutions. Solutions for a panel with left-edge stiffener and a centered fastener hole are plotted in Fig. 38. The results for K_I and K_{II} appear to be symmetrical. In fact, very little difference can be observed between the unstiffened, symmetrically stiffened and asymmetrically stiffened panels (compare Figs. 35, 37 and 38). Some difference is noted when the fastener hole is offset 1.5 inches to the right, away from the stiffened edge. Figure 39 presents the results for this case, which are almost identical to the results for an unstiffened panel (compare with Fig. 36). The tentative conclusion from these results is that edge stiffeners and moderate offsets do not appreciably affect K_I and K_{II} for cracks at the fastener hole, at least for uniform tension loading and when the hole and crack are not extremely close

to an edge or stiffener.

Figure 40 summarizes the dimensional information for a panel model used to study a broad range of a/R_I ratios with W/D_O fixed at 1.59. Table 3 summarizes the results for a series of cases with equal-length cracks at $\theta=0,\pi$. These may be compared directly with handbook data because the dimensions of the model correspond precisely with a published curve. The results are seen to be quite reasonable over the entire range:

$$0.05 \leq a/R_I \leq 1.25$$

Indeed some of the error indicated in the table may be attributed to inaccuracy in reading the handbook chart. A number of other cases for a single crack at $\theta=0$ and for two unequal-length cracks are compared with the foregoing results in Table 4. Complete butterfly plots of the K_I and K_{II} data computed from these runs will appear in a later report.

Section 6

CONCLUSIONS

This report has traced the development of a parametric finite-element analysis program for computation of Mode I and Mode II stress intensity factors in stiffened and unstiffened panels with cracks emanating from centered and offset fastener holes. Early attempts to model the structure entirely with conventional elements, except for the crack-tip element and a few mesh-expander elements, proved to be unsuccessful. Test runs of these types of finite-element models quickly demonstrated that asymmetric grading of the mesh was forced by the need to accommodate an offset hole and at the same time to keep the size of the model within economic bounds. The asymmetric mesh was found to give extremely poor results for computed displacements in regions of low stress gradient, and was therefore abandoned.

The assumed-stress hybrid method was called upon once again, this time to provide a special element which could handle the transition from circular geometry near the fastener hole to the cartesian geometry natural to the rest of the panel. As finally developed, the hybrid element occupied a 45-degree sector between its inner (circular) and outer boundary. Airy stress functions from classical elasticity solutions near a hole in an infinite plate were used to provide the assumed stress field. The new

element allowed a much cleaner, more economical finite-element mesh to be designed, while at the same time providing accurate computations near the fastener hole. Performance tests have shown that the element is capable of occupying a stress-free fastener hole boundary directly, and of accepting interference-fit and cosine bearing loading on the fastener hole surface. Additional tests demonstrated that conventional quadrilateral elements could be coupled in rings inside the inner boundary of the new hybrid element, with no loss of accuracy. Some tests were also conducted with hybrid crack-containing elements replacing some of the quadrilaterals. These tests demonstrated that stress intensity factors could be computed to within a few percent of the values given by well-established independent solutions based on classical methods. In the final phase of the project, a parametric program was developed and verified for analysis of tension panels in the various configurations mentioned above. During the verification tests, the hybrid fastener hole-ring combination was subjected to a variety of dimensional and shape parameters to further extend the range of measured performance. The results of these tests have demonstrated that the panel program is capable of computing stress intensity factors to within 5 percent or better for fastener hole and crack sizes found in current airframes.

The many example results presented in this report are still a rather limited data base, when compared with the number of stress

intensity factor solutions needed for a comprehensive designer's handbook. Some additional data, which were generated in the final verification tests but not included in this report, will appear in a later report in this series. However, further verification tests are still required to broaden the range of applicability of the panel program. One concern which has not yet been answered is how close a row of fastener holes may be spaced, or how near may a fastener hole approach the edge of a panel, before the finite-element model experiences unacceptable degradation of accuracy. The data generated thus far seem to indicate that there will be a limit in this respect, and that the limit may be severe. Addition of mid-edge nodes to the edges which span between the hybrid fastener hole element's inner and outer boundaries may permit closer spacing, but will also require additional terms in the assumed stress field.

Another continuing concern is associated with the verification process itself. The combination of crack-containing and fastener hole elements gives the capability to compute K_I and K_{II} for so many varied configurations that the numerical analyst finds himself sailing in uncharted waters. At the present time, parametric codes like the panel program can only be calibrated against classical solutions for K_I at one or two data points, while they may compute as many as 100 data points each for K_I and K_{II}.

REFERENCES

1. Orringer, O., "Fracture Mechanics Analysis of an Attachment Lug", Aeroelastic and Structures Research Laboratory, MIT, ASRL TR 177-1, AFFDL-TR-75-51, December 1974.

2. Lasry, S.J., "Derivation of Crack Element Stiffness Matrix by the Complex Variable Approach", Aeroelastic and Structures Research Laboratory, MIT, ASRL TR 170-2, AFOSR-TR-73-1602, February 1973.

3. Lin, K.Y., Tong, P. and Orringer, O., "Effect of Shape and Size on Hybrid Crack-Containing Finite Elements", to be presented at ASME Pressure Vessels and Piping Conference, San Francisco, California, June 1975.

4. Luk, C.H., "Assumed Stress Hybrid Finite Element Methodology for Fracture Mechanics and Elastic-Plastic Analysis", Aeroelastic and Structures Research Laboratory, MIT, ASRL TR 170-1, December 1972.

5. Tong, P., Pian, T.H.H. and Lasry, S.J., Int J Num Meth Eng, Vol. 7, pp. 297-308 (1973).

6. Orringer, O. and French, S.E., "FEABL (Finite Element Analysis Basic Library) User's Guide", Aeroelastic and Structures Research Laboratory, MIT, ASRL TR 162-3, AFOSR-TR-72-2228, August 1972. (Supplemented by ASRL TM 162-3-3 and ASRL TM 162-3-4, February 1974.)

7. Dally, J.W. and Riley, W.F., Experimental Stress Analysis, McGraw-Hill, 1964.

8. Abramowitz, M. and Stegun, I.A. (ed.), Handbook of Mathematical Functions, National Bureau of Standards, U.S. Dept. of Commerce, Ninth Ed., November 1970.

9. Timoshenko, S. and Goodier, J.N., Theory of Elasticity, McGraw-Hill, 1961.

10. Paris, P.C. and Sih, G.C., "Stress Analysis of Cracks", ASTM STP 381, Fracture Toughness Testing and Its Applications, 1965, pp. 30-83.

11. Tada, H., Paris, P.C. and Irwin, G.R., The Stress Analysis of Cracks Handbook, DEL Research Corporation, Hellertown Pa., 1973.

TABLE 1

COMPARISON OF CLASSICAL AND FINITE-ELEMENT FRACTURE MECHANICS SOLUTIONS

Number of Cracks	Crack Size, a (in.)	Hole Radius, R_I (in.)	a/R_I	K_I (psi$\sqrt{in.}$) Exact	FEM	% Error
2	0.04	0.08	0.5	650	630	3.0
1	0.04	0.10	0.4	660	675	2.4
2	0.08	0.10	0.8	792	771	2.7
2	0.05	0.20	0.25	910	910	---
2	0.20	0.20	1.0	1150	1170	1.9

TABLE 2

COMPARISON OF CLASSICAL AND FINITE-ELEMENT RESULTS FROM THREE DEMONSTRATION EXAMPLES

(Fastener hole with 2 cracks)

Case	Fig.	Crack Size (in.)	Hole Rad. (in.)	a/R_I	HOLEL W/D_O	K_I (psi$\sqrt{in}$) Exact	FEM	% Error
1	33	0.3	0.5	0.6	1.59	[illegible]	1900	1.0
2	34	0.125	0.25	0.5	3.17	[illegible]	1134	4.1
3	35	0.125	0.125	1.0	6.35	886	880	0.7

TABLE 3

PERFORMANCE OF PANEL PROGRAM AS A FUNCTION OF a/R_I RATIO

(Structure model in Fig. 40; cracks at $\theta=0,\pi$.)

a(in.) and a/R_I	K_I (psi$\sqrt{in.}$) Exact*	FEM	% Error
0.05	1144	1160	1.4
0.1	1673	1613	3.6
0.2	2037	2051	0.7
0.4	2474	2445	1.2
0.6	2735	2715	0.7
0.8	2973	2897	2.6
1.0	3257	3099	4.9
1.2	3521	3401	3.4
1.25	3588	3500	2.5

*Ref. 11, p. 19.4, curve marked h/b=2, R/b=0.25.

TABLE 4

ADDITIONAL TEST OF a/R_I RATIO

(Structure model in Fig. 40; one crack or two unequal cracks.

a(in.) and a/R_I		K_I (psi$\sqrt{in.}$) at $\theta=0$		%
$\theta=0$	$\theta=\pi$	FEM	Exact (from Table 3)	Difference
0.05	---	1155	1144	1.0
0.1	---	1600	1673	4.4
0.2	---	2000	2037	1.8
0.4	---	2302	2474	6.9
0.6	---	2466	2735	9.8
0.8	---	2522	2973	15.2
1.0	---	2602	3257	20.1
1.2	---	2758	3521	21.7
1.25	---	2813	3588	21.6
0.05	1.25	1447	1144	26.5
1.25	0.05	2824	3588	21.3

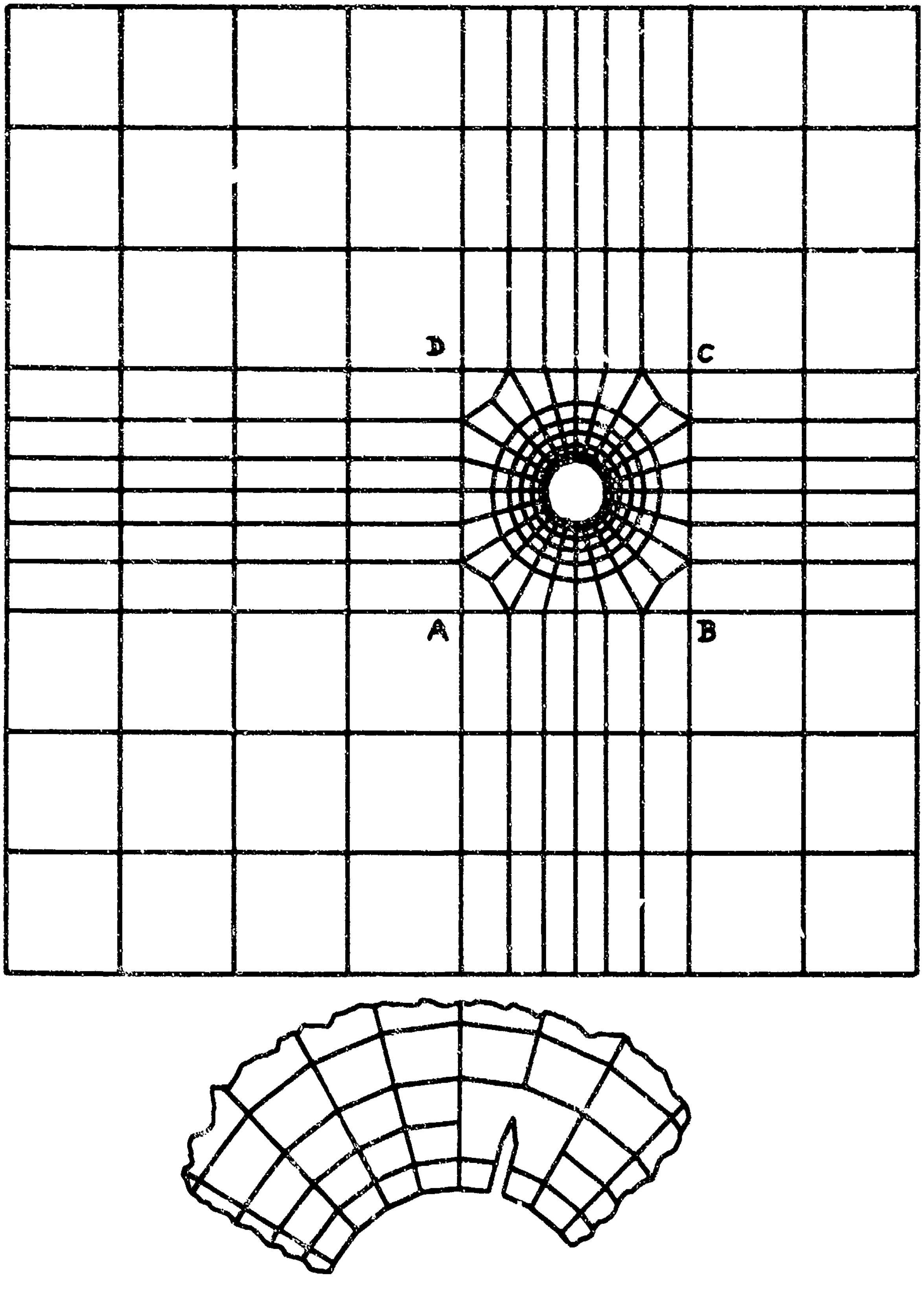


FIG. 1 CONVENTIONAL FINITE-ELEMENT MODEL OF PANEL WITH OFFSET FASTENER HOLE

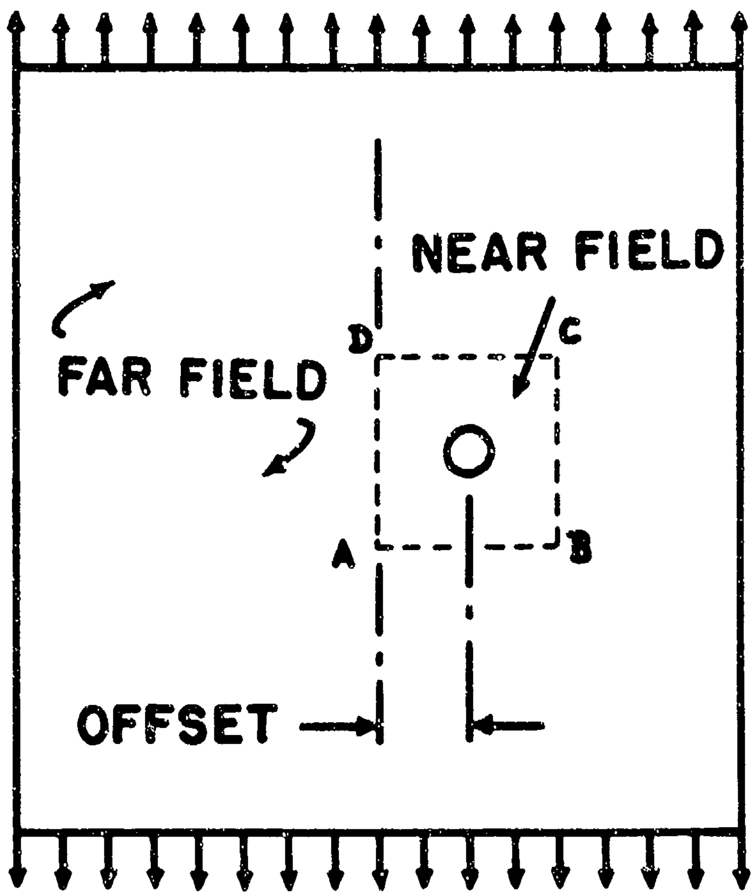


FIG. 2 SUBDIVISION OF PANEL INTO NEAR-FIELD AND FAR-FIELD REGIONS

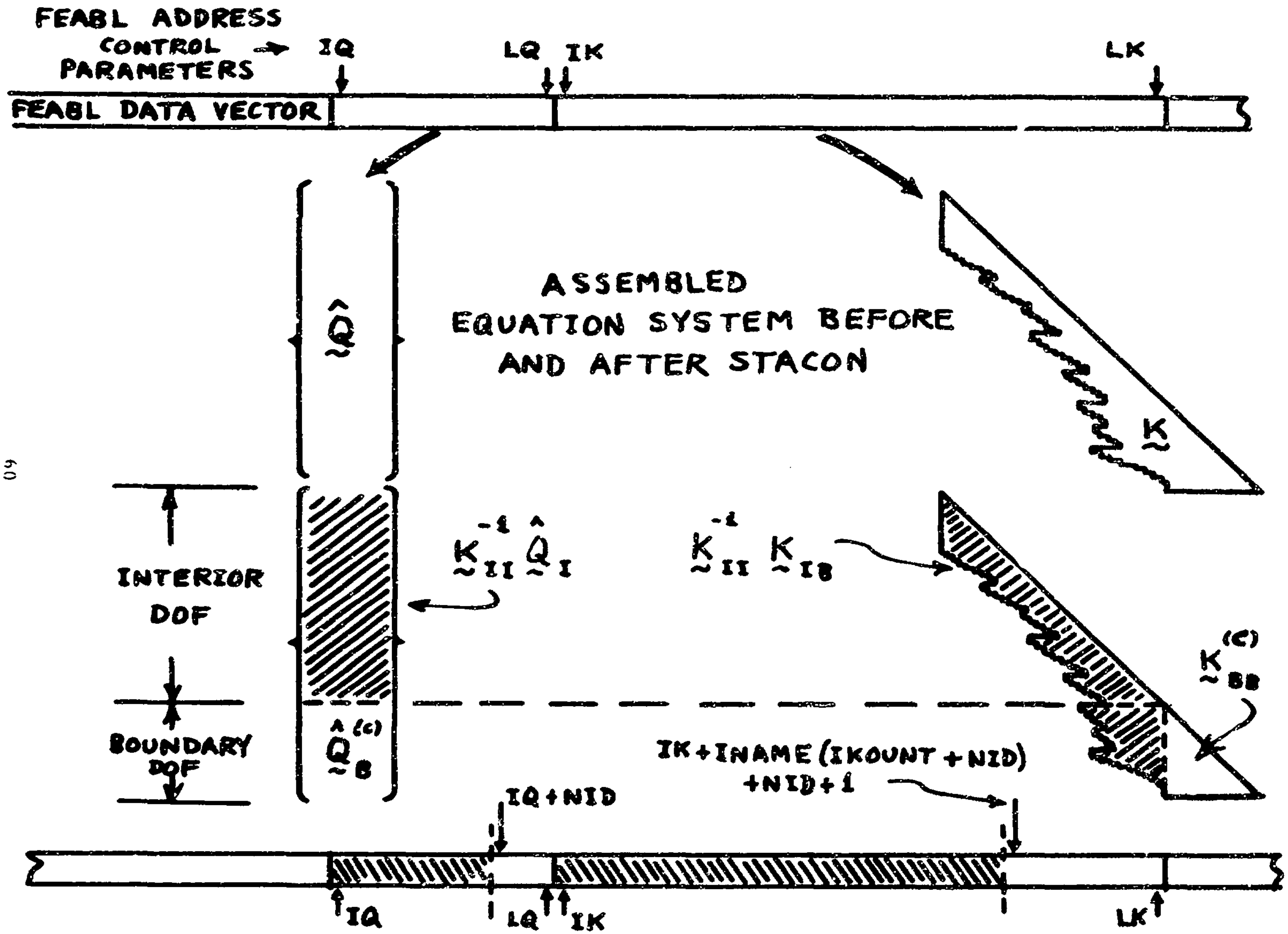


FIG. 3 ACTION OF FEABL-2 SUBROUTINE STACON ON ASSEMBLED SUBSTRUCTURE STIFFNESS MATRIX

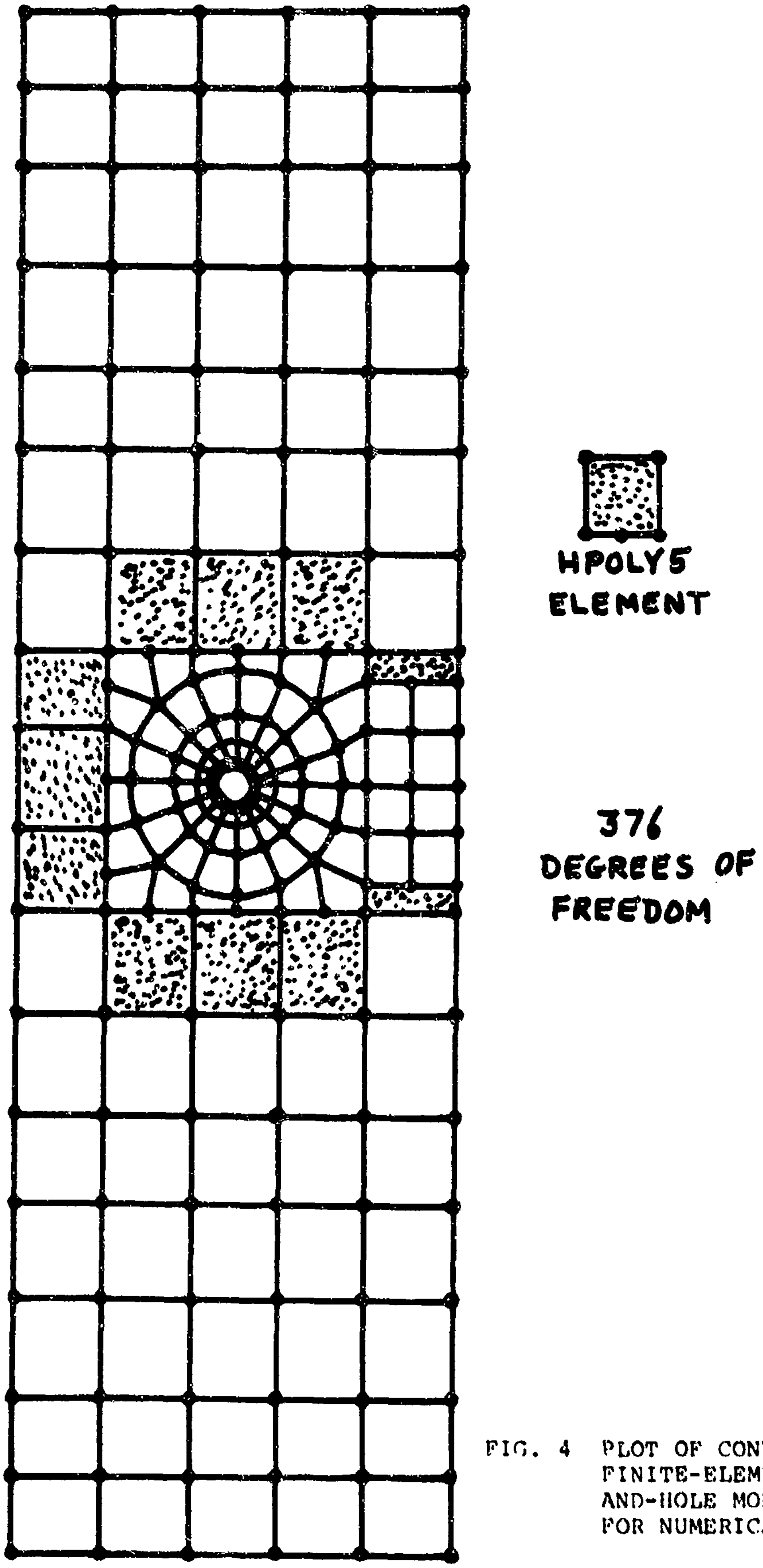


FIG. 4 PLOT OF CONVENTIONAL FINITE-ELEMENT PANEL-AND-HOLE MODEL TESTED FOR NUMERICAL ACCURACY

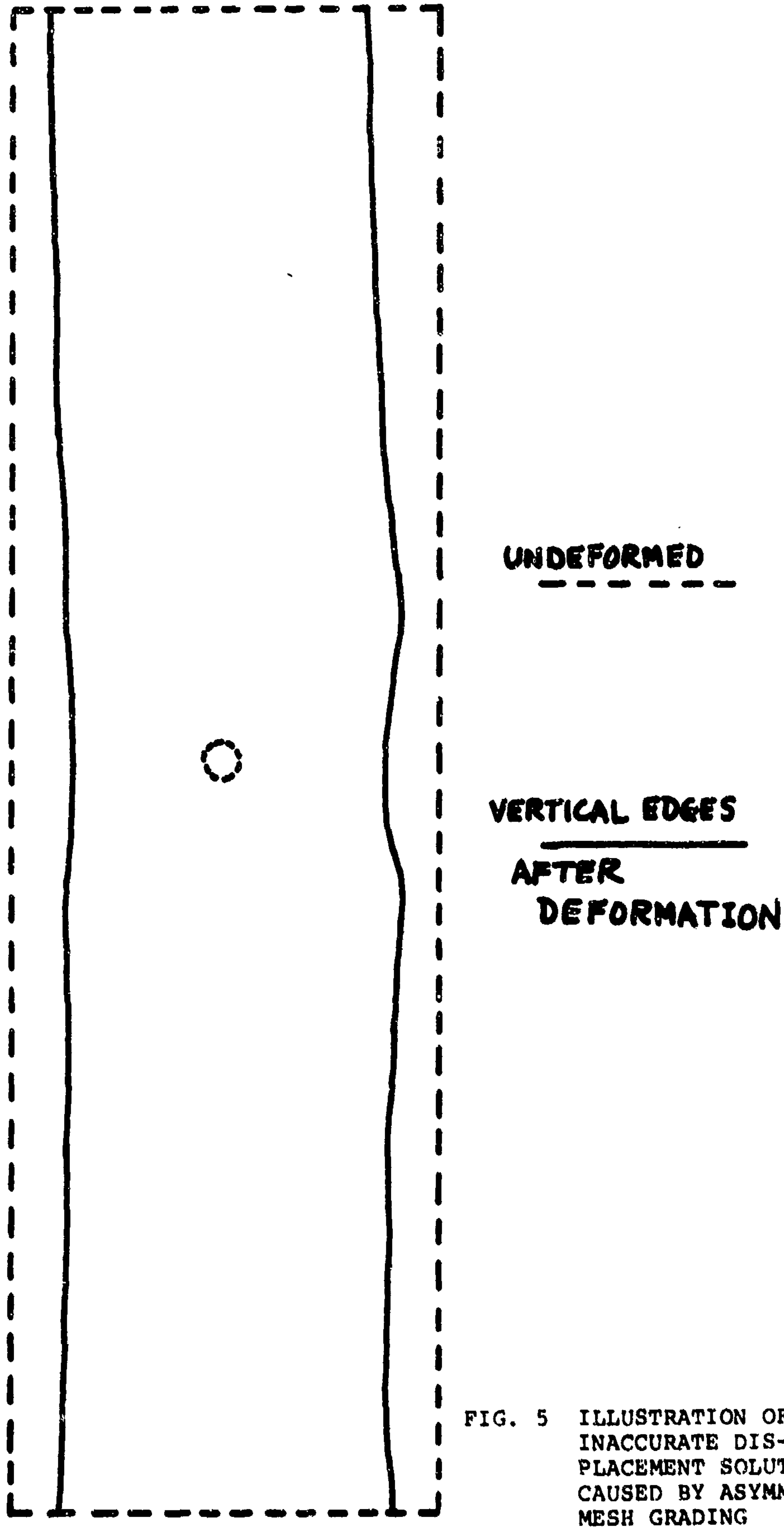


FIG. 5 ILLUSTRATION OF INACCURATE DISPLACEMENT SOLUTION CAUSED BY ASYMMETRIC MESH GRADING

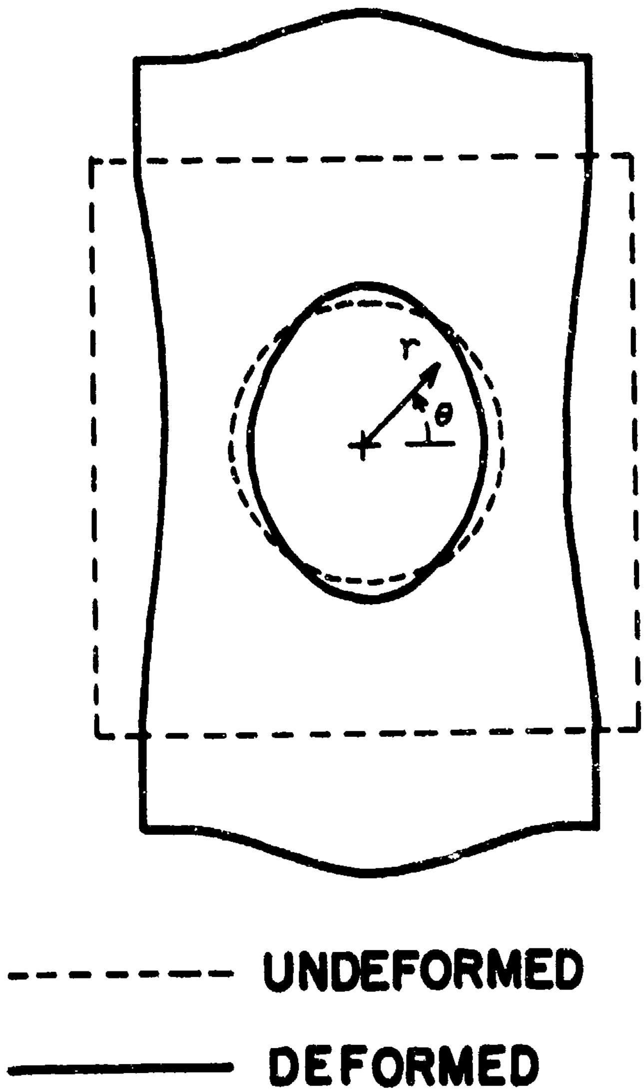


FIG. 6 EXPECTED DEFORMATION BEHAVIOR OF NEAR-FIELD REGION

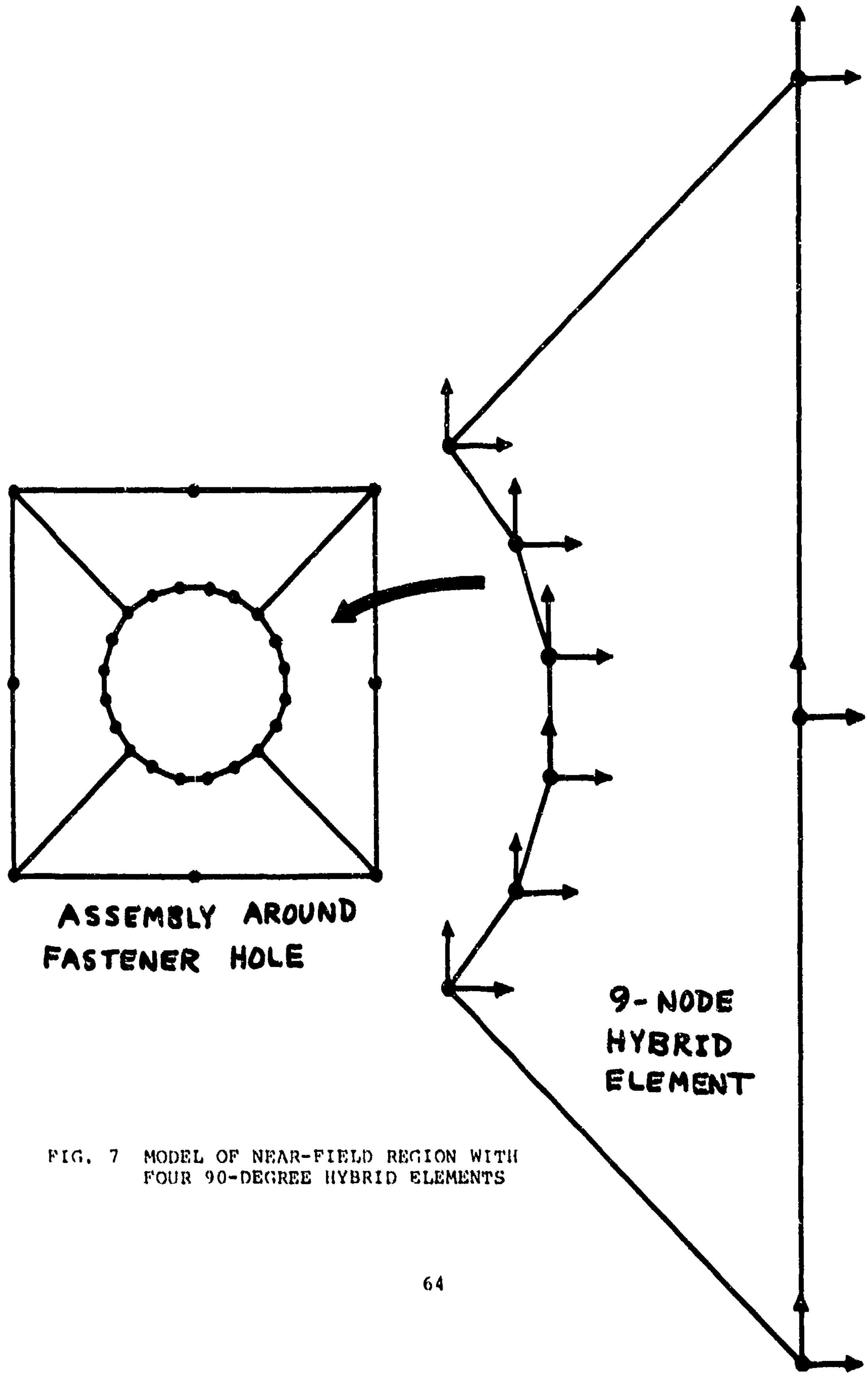


FIG. 7 MODEL OF NEAR-FIELD REGION WITH FOUR 90-DEGREE HYBRID ELEMENTS

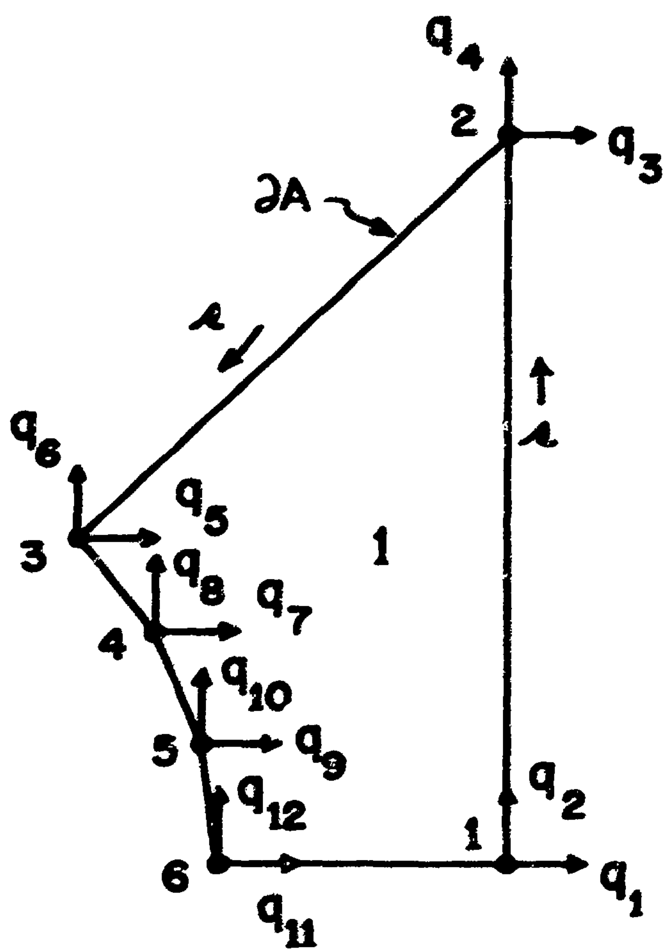


FIG. 8 LOCAL NUMBERING CONVENTION FOR 45-DEGREE HYBRID ELEMENT

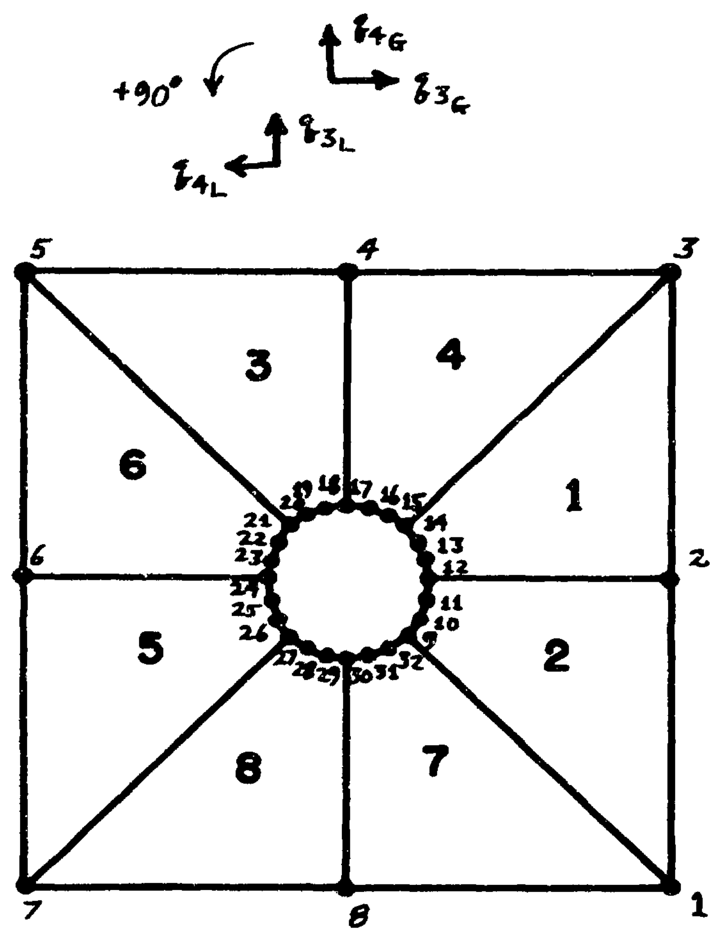


FIG. 9 MODEL OF NEAR-FIELD REGION WITH EIGHT 45-DEGREE HYBRID ELEMENTS

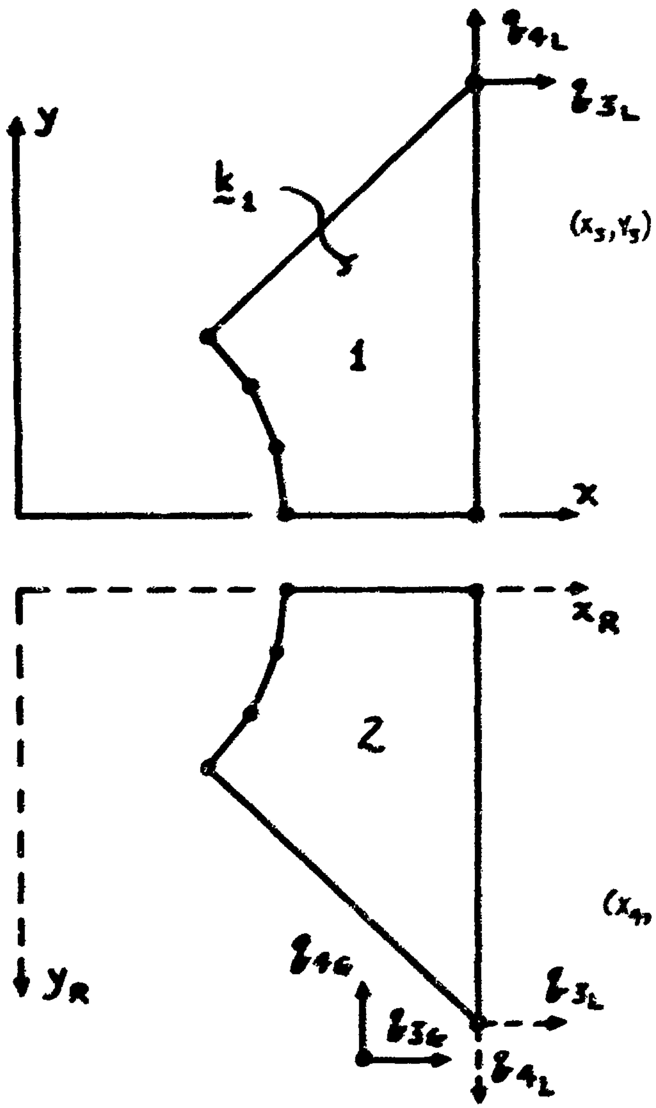


FIG. 10 EXAMPLE OF REFLECTION TRANSFORMATION

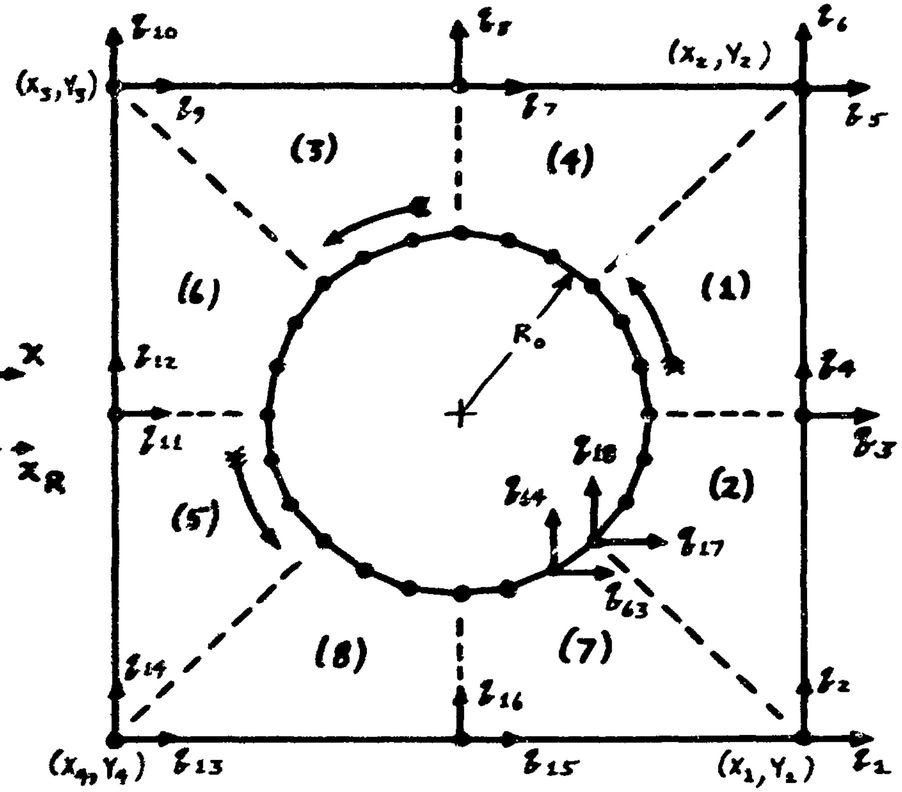


FIG. 11 LOCAL NUMBERING CONVENTION FOR COMPLETED HYBRID FASTENER HOLE ELEMENT (HOLEL)

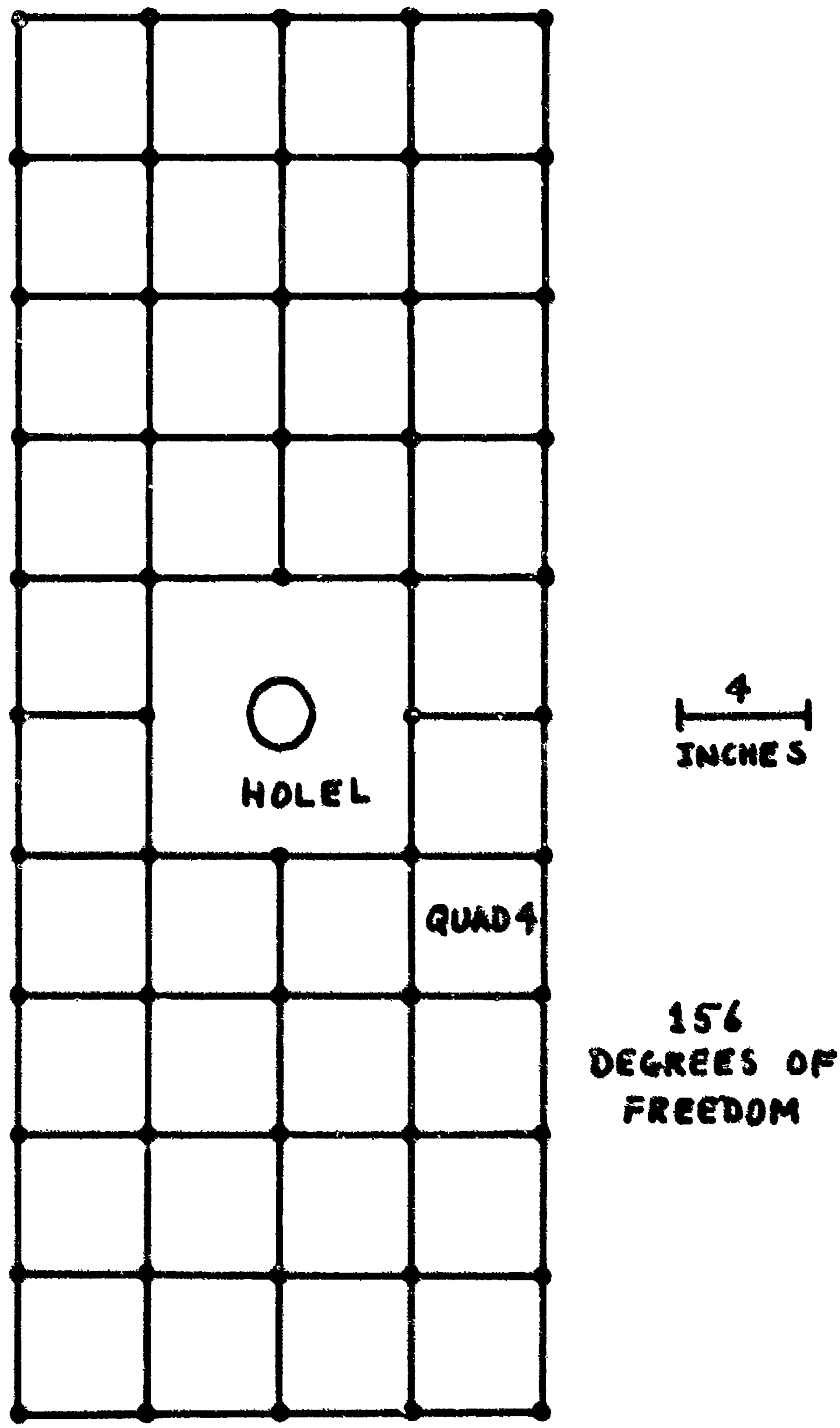


FIG. 12 FINITE-ELEMENT MODEL FOR PERFORMANCE TESTING OF HOLEL

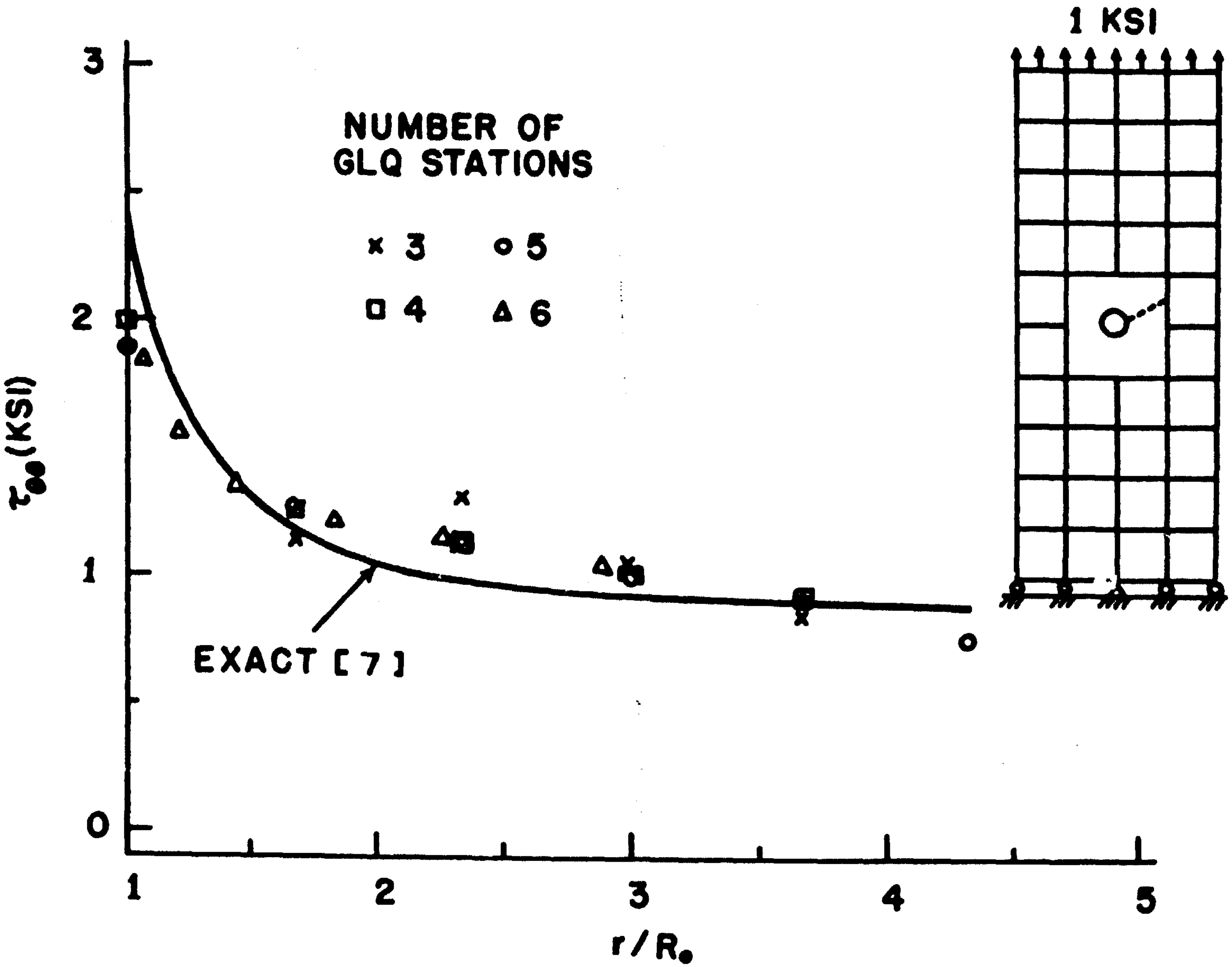


FIG. 13 SENSITIVITY OF HOLEL TO NUMBER OF GLQ INTEGRATION STATIONS

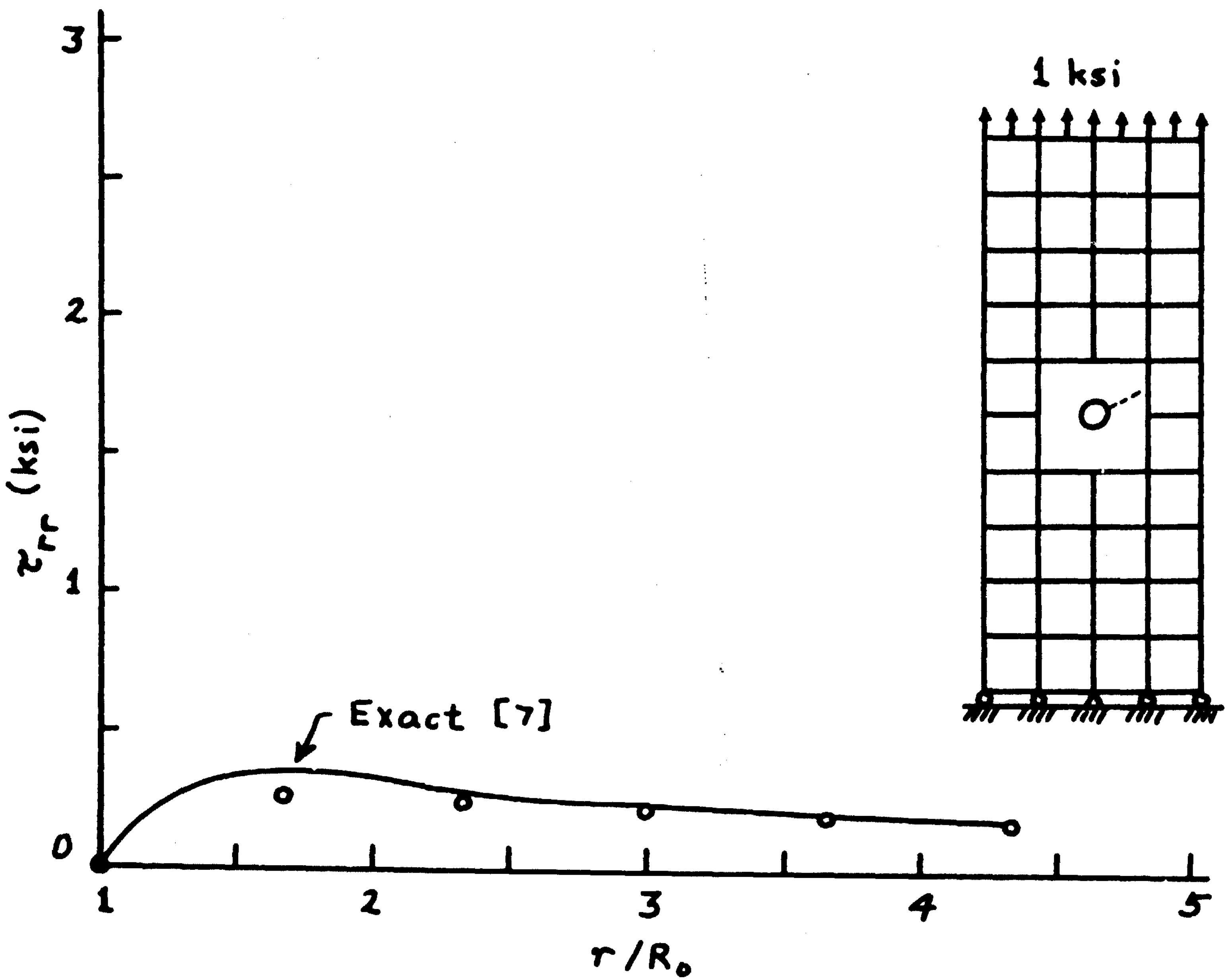


FIG. 14 BEHAVIOR OF τ_{rr} NEAR STRESS-FREE EDGE

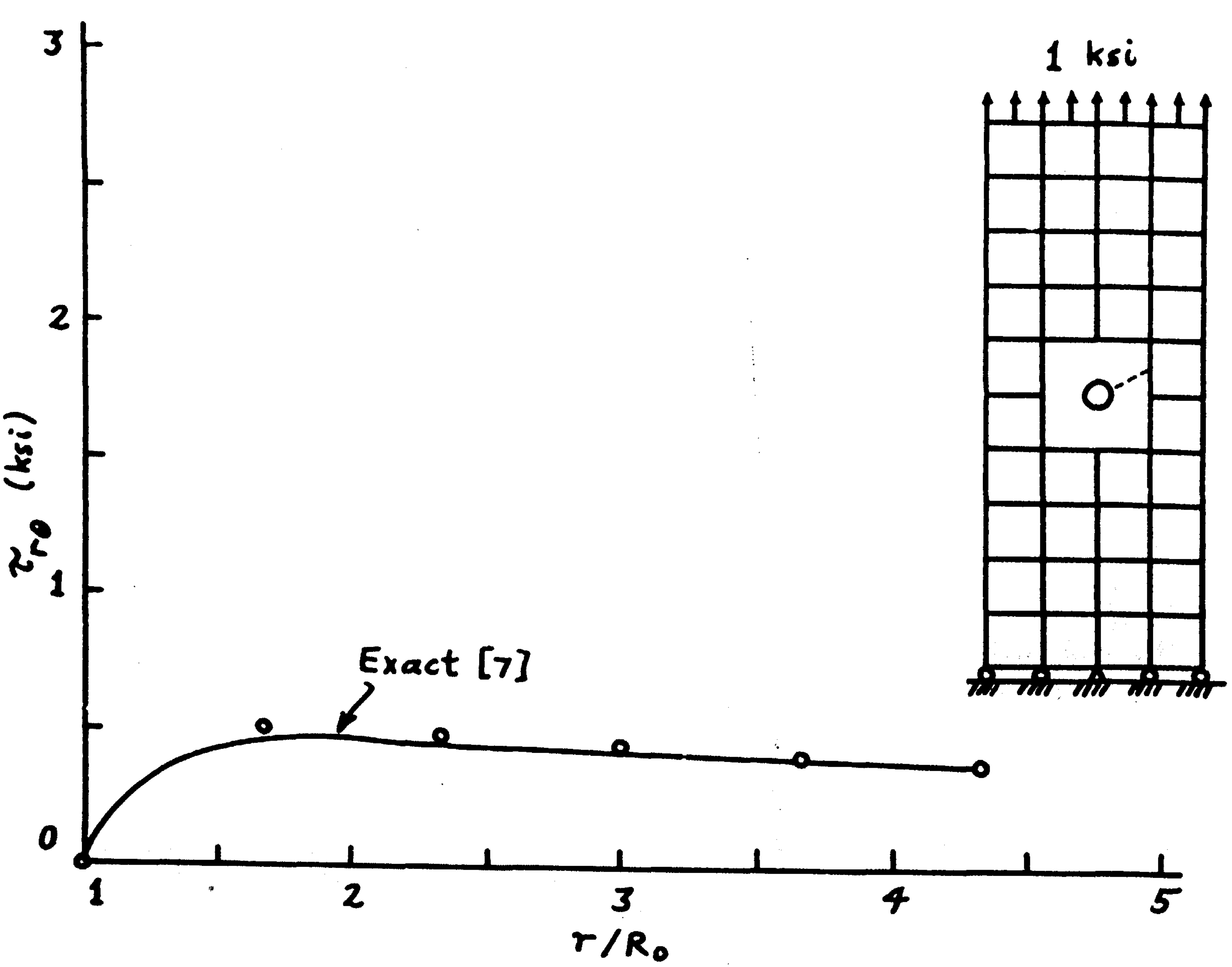


FIG. 15 BEHAVIOR OF $\tau_{r\theta}$ NEAR STRESS-FREE EDGE

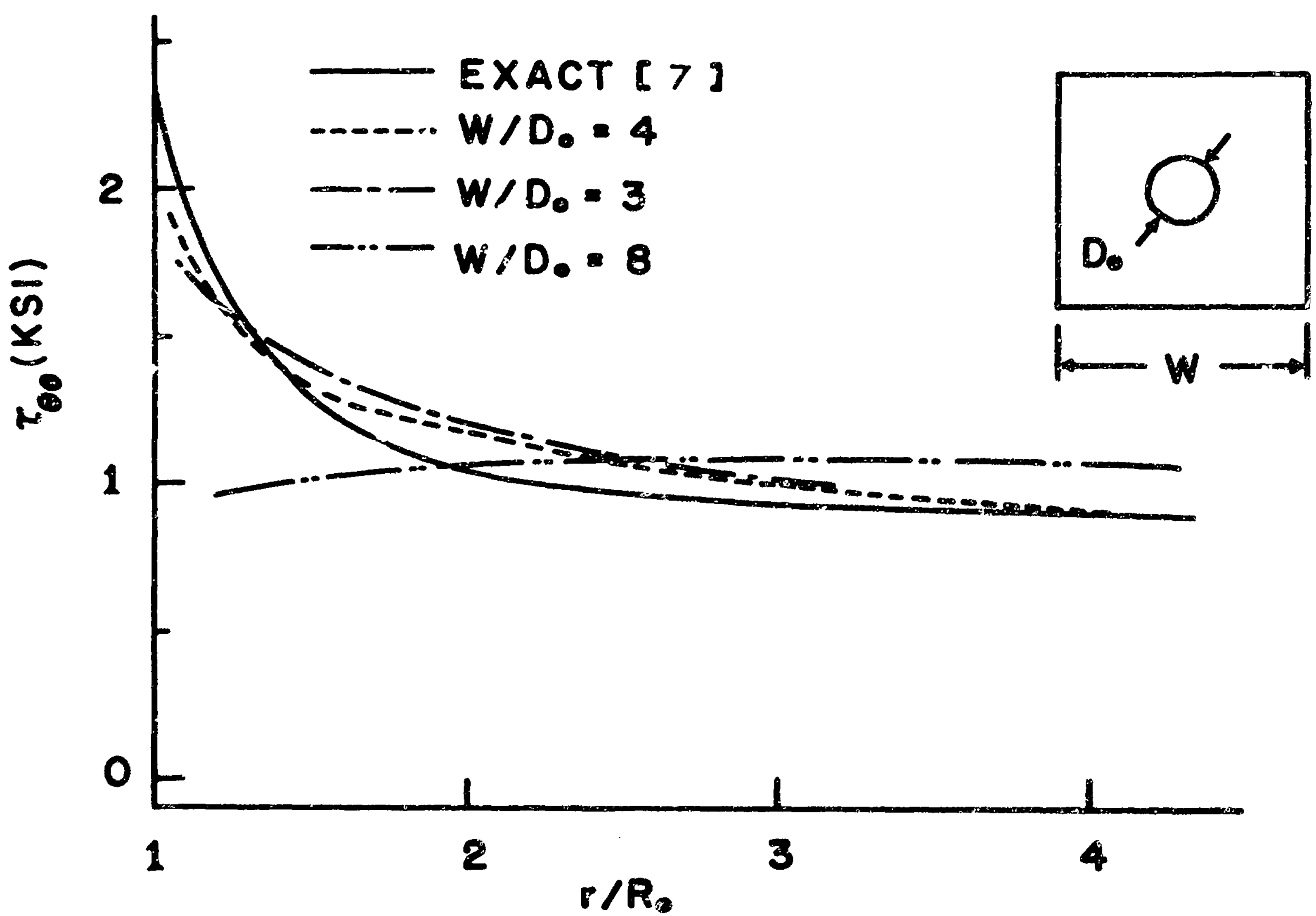


FIG. 16 SENSITIVITY OF HOLEL TO SHAPE PARAMETER W/D_o

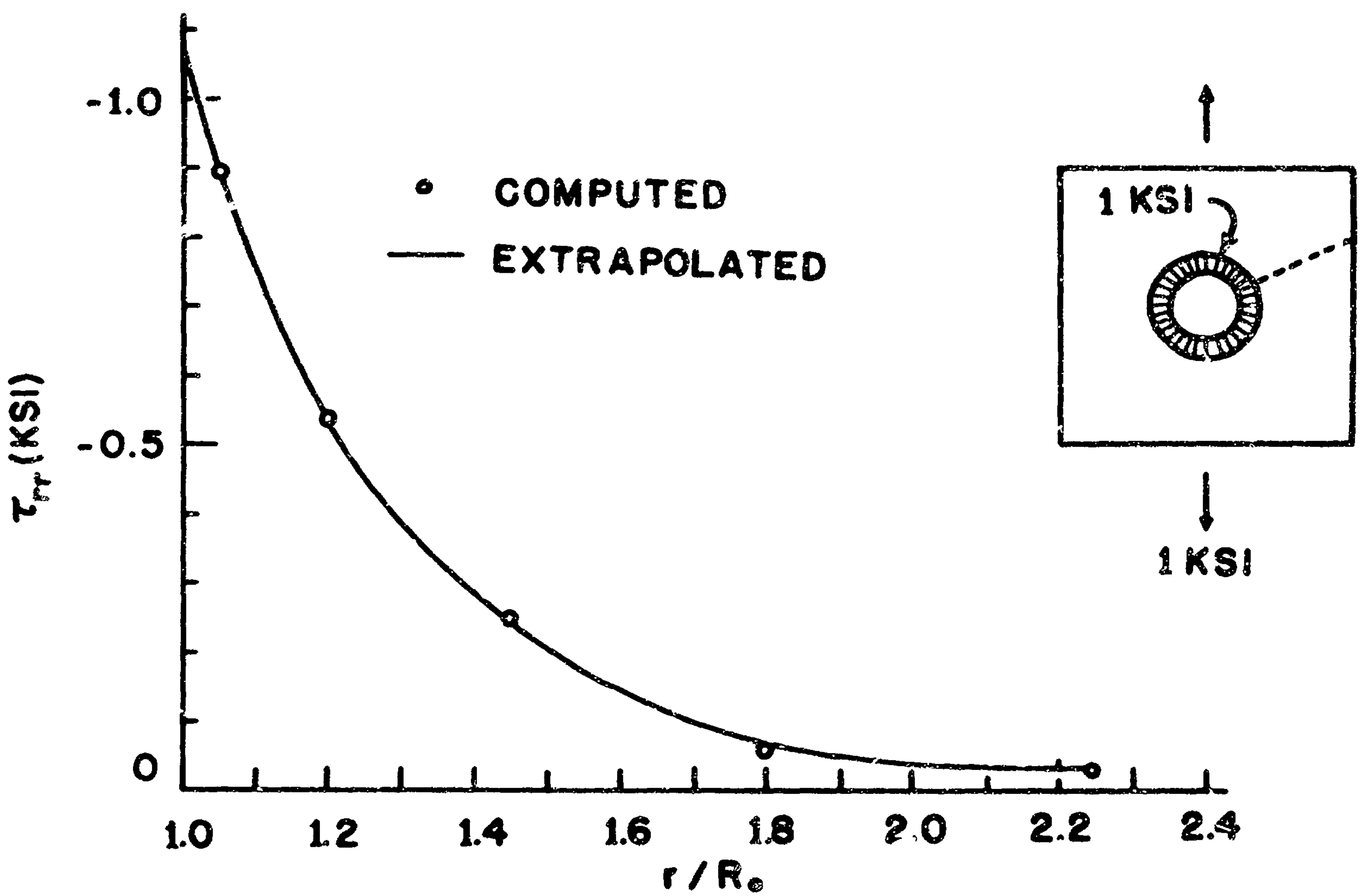


FIG. 17 ABILITY OF HOLEL TO ACCEPT AN INTERFERENCE FIT

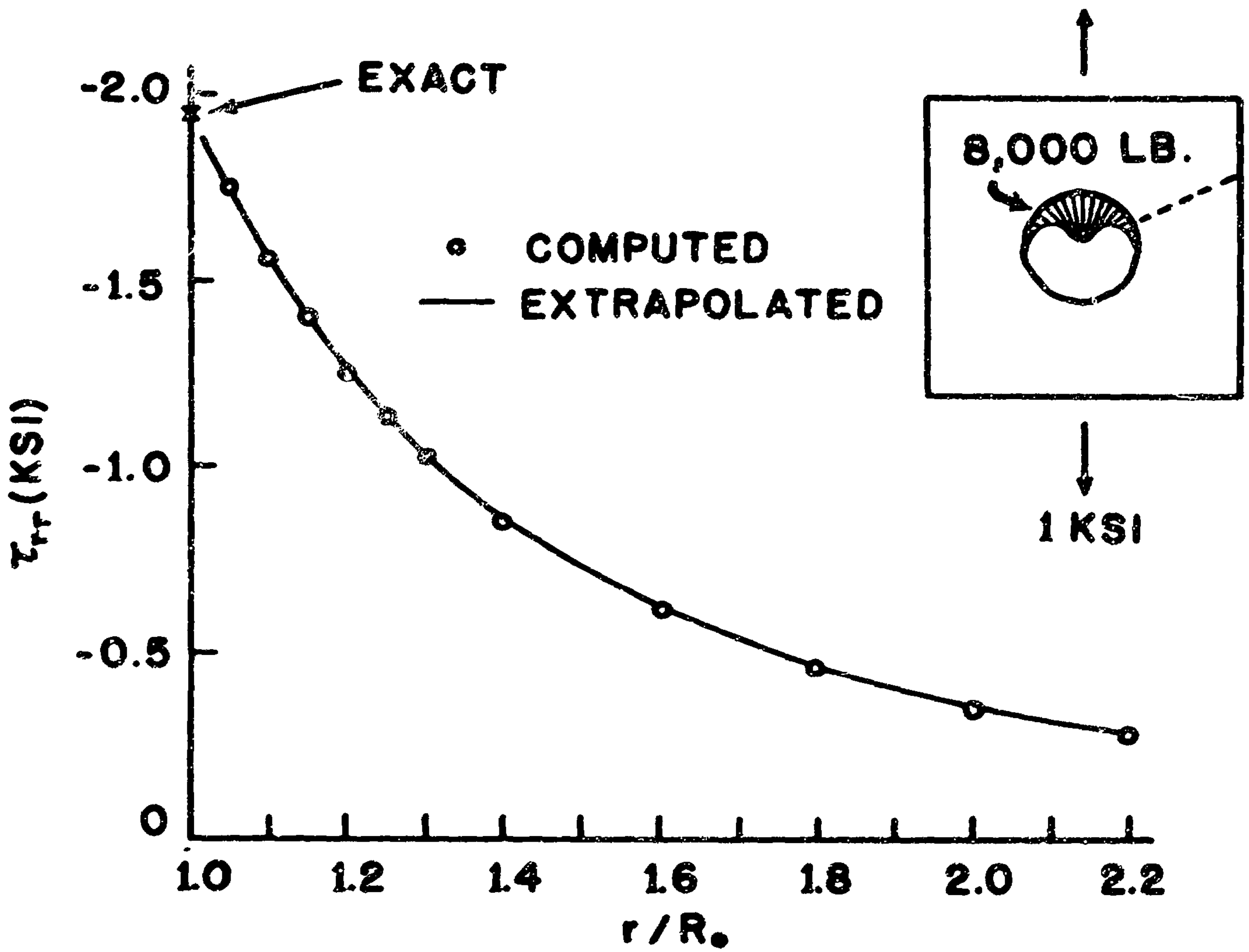


FIG. 18 ABILITY OF HOLEL TO ACCEPT A BEARING LOAD

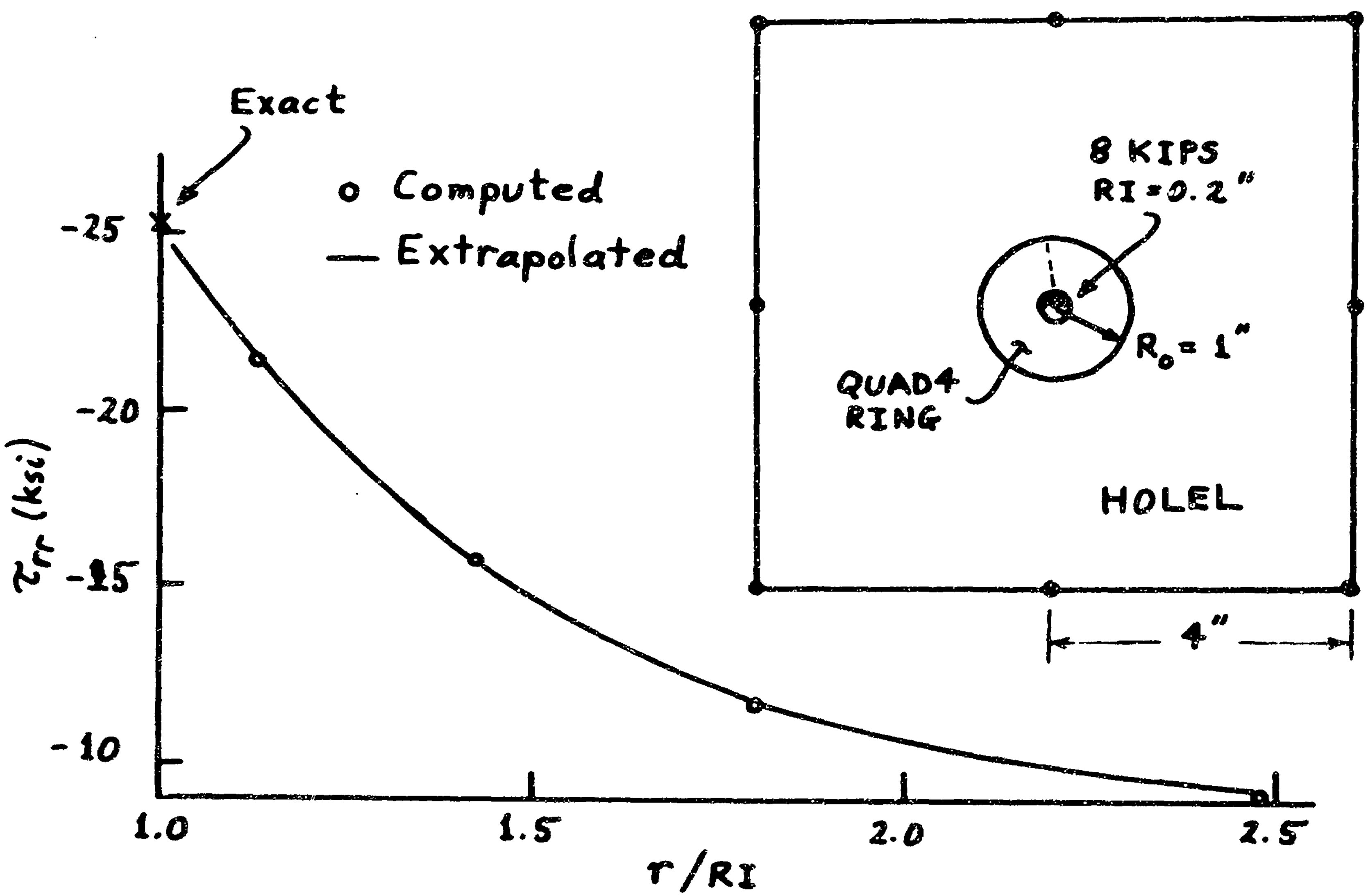


FIG. 19 TEST OF HOLEL WITH INTERIOR RING SUBJECTED TO BEARING LOAD

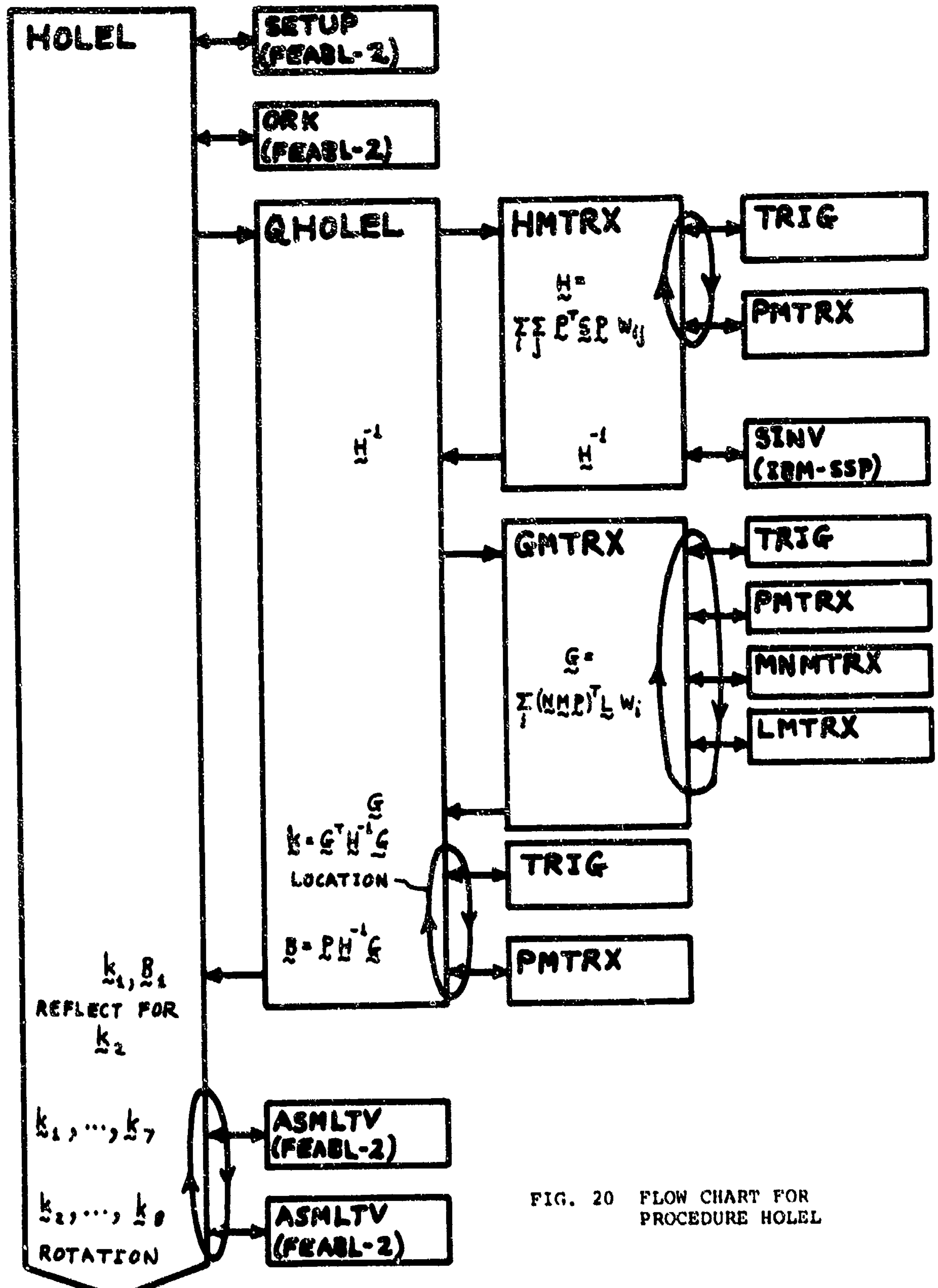


FIG. 20 FLOW CHART FOR PROCEDURE HOLEL

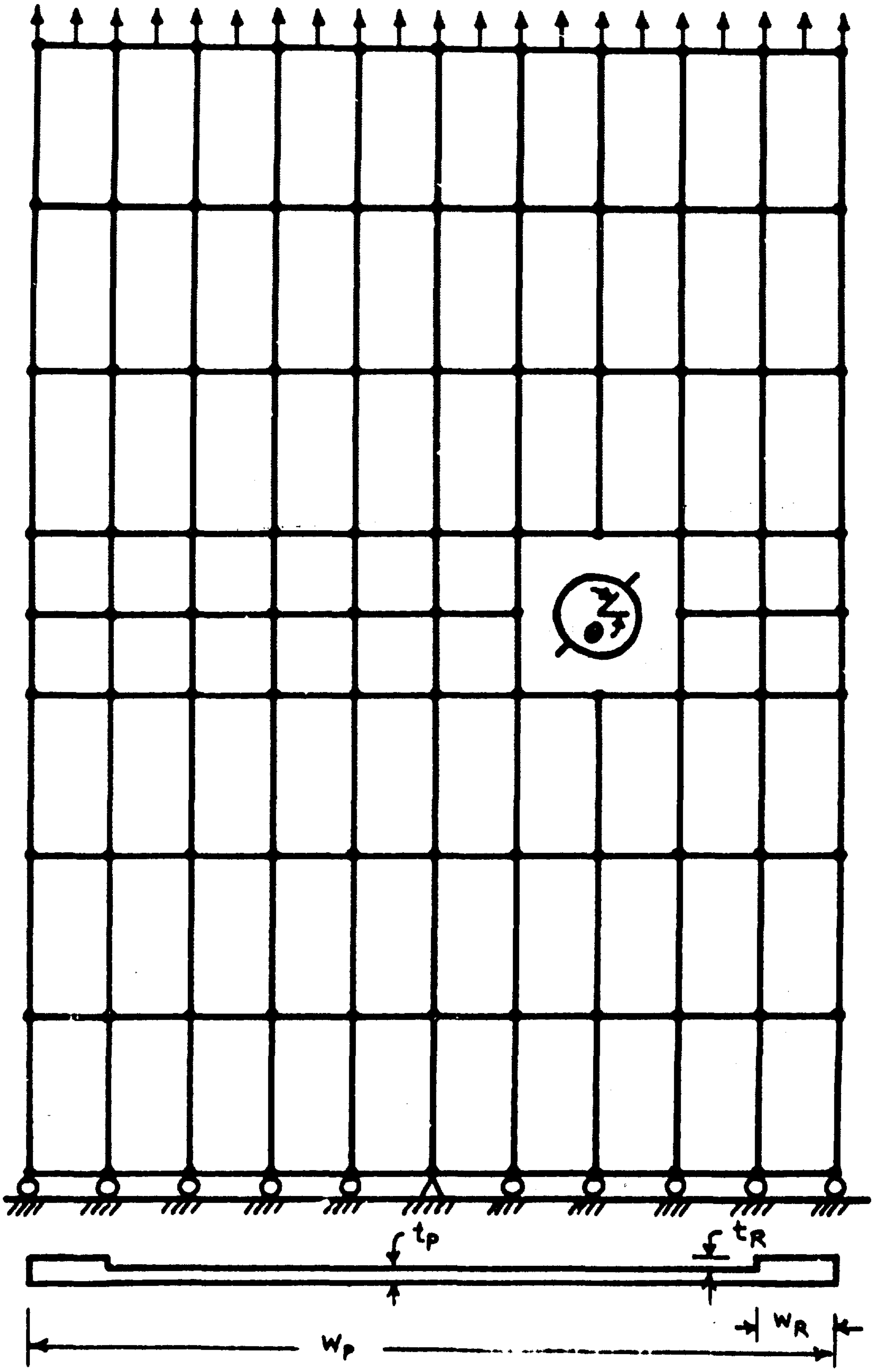


FIG. 21 PANEL PROGRAM STRUCTURE

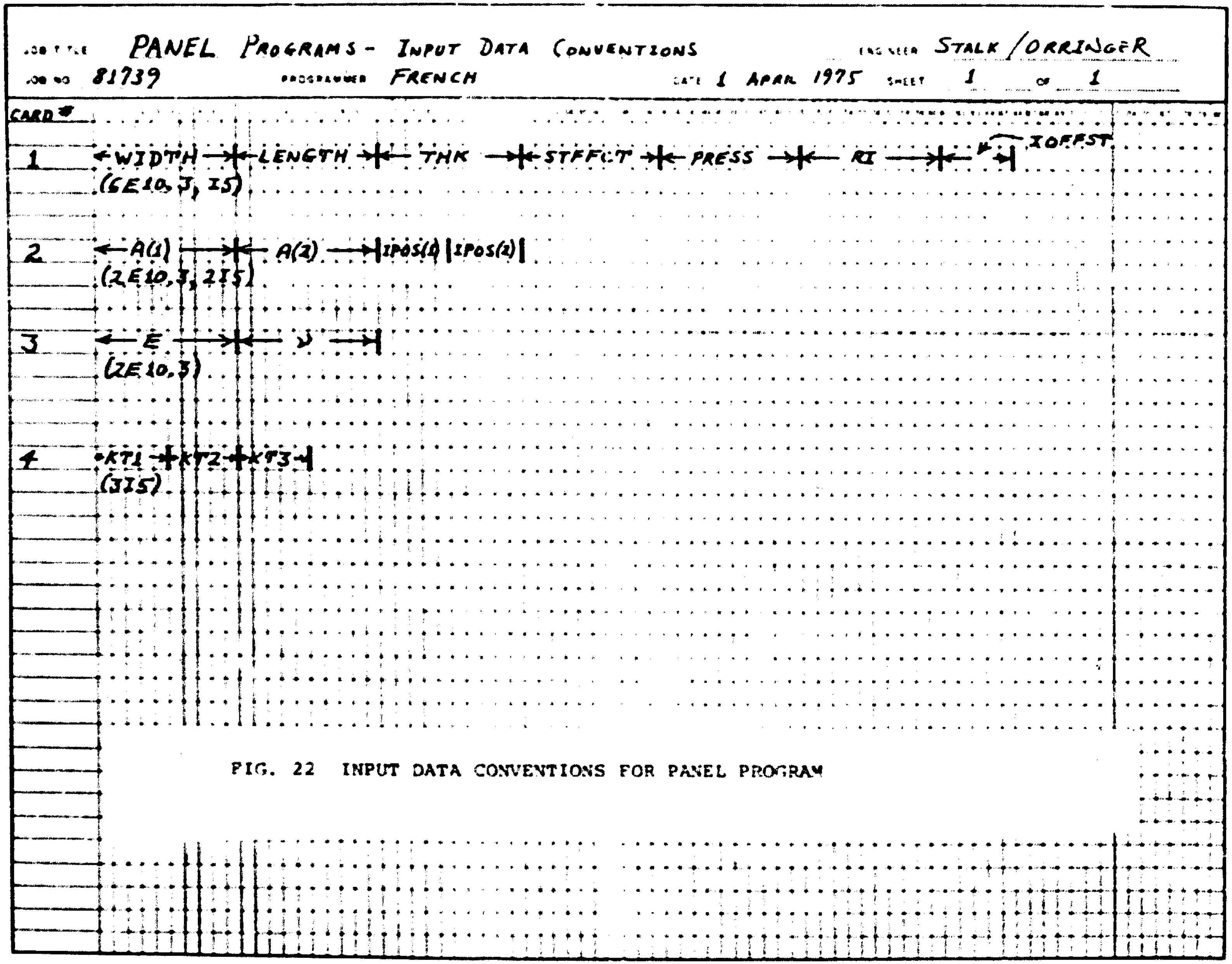


FIG. 22 INPUT DATA CONVENTIONS FOR PANEL PROGRAM

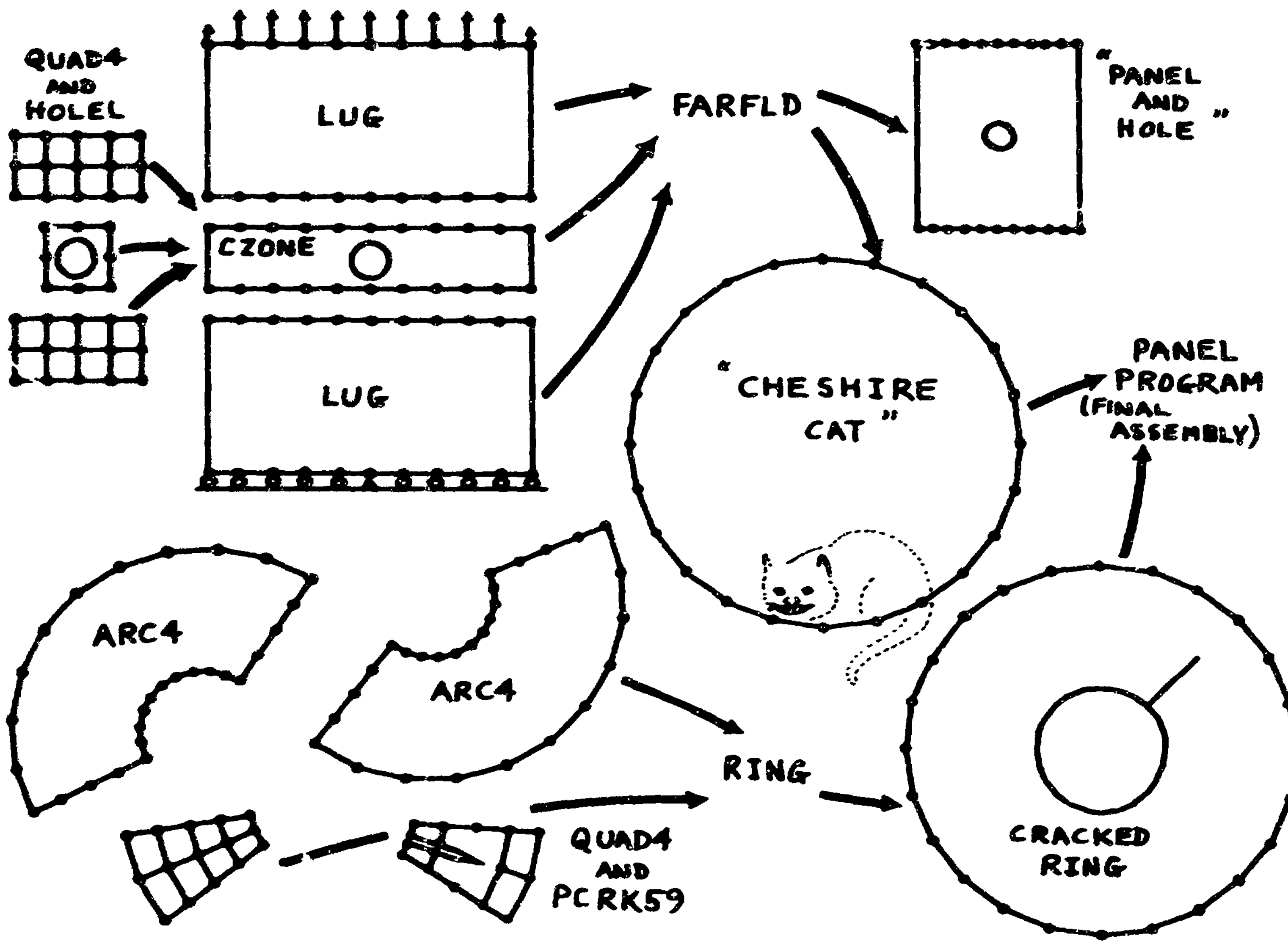
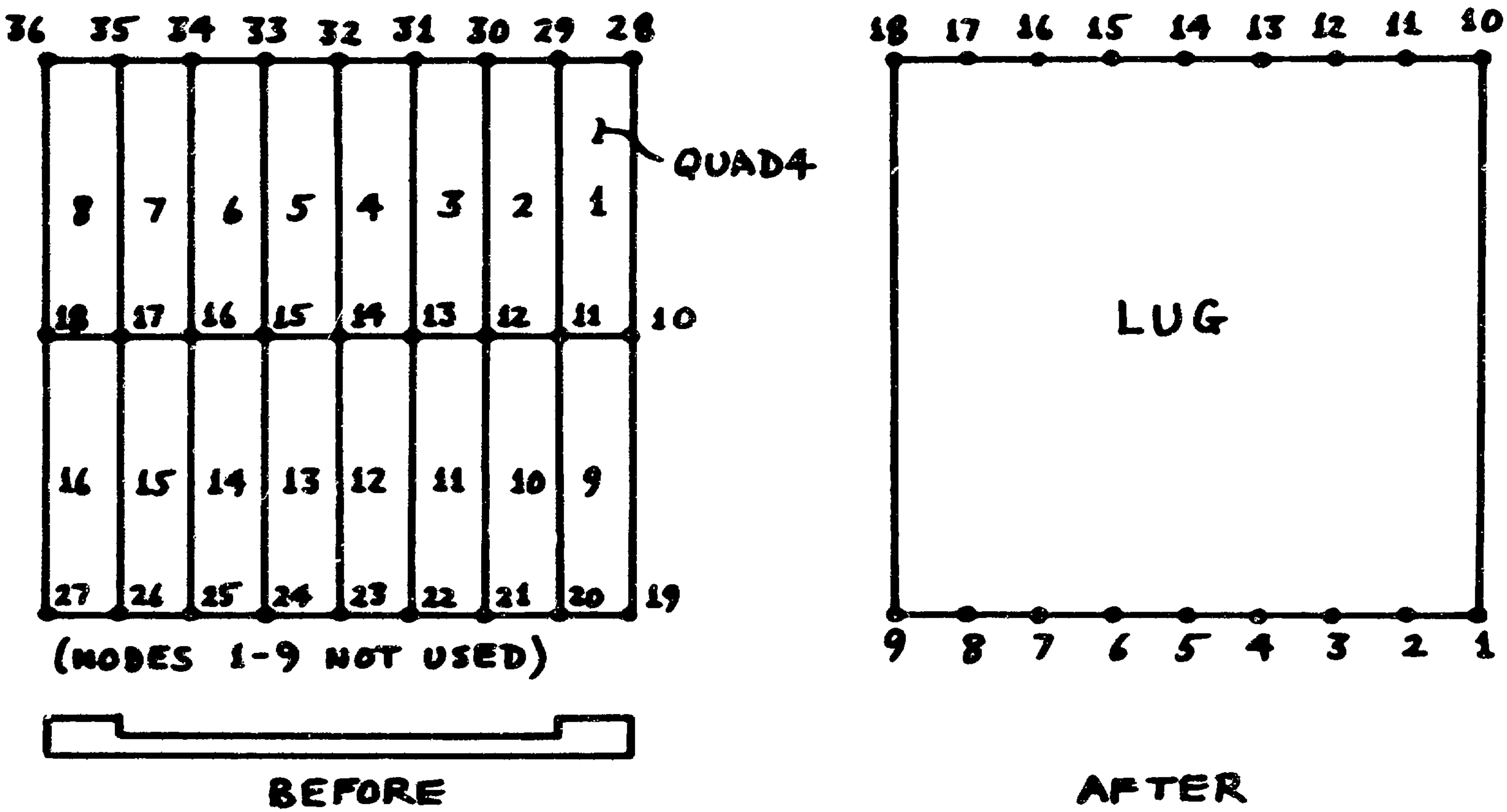


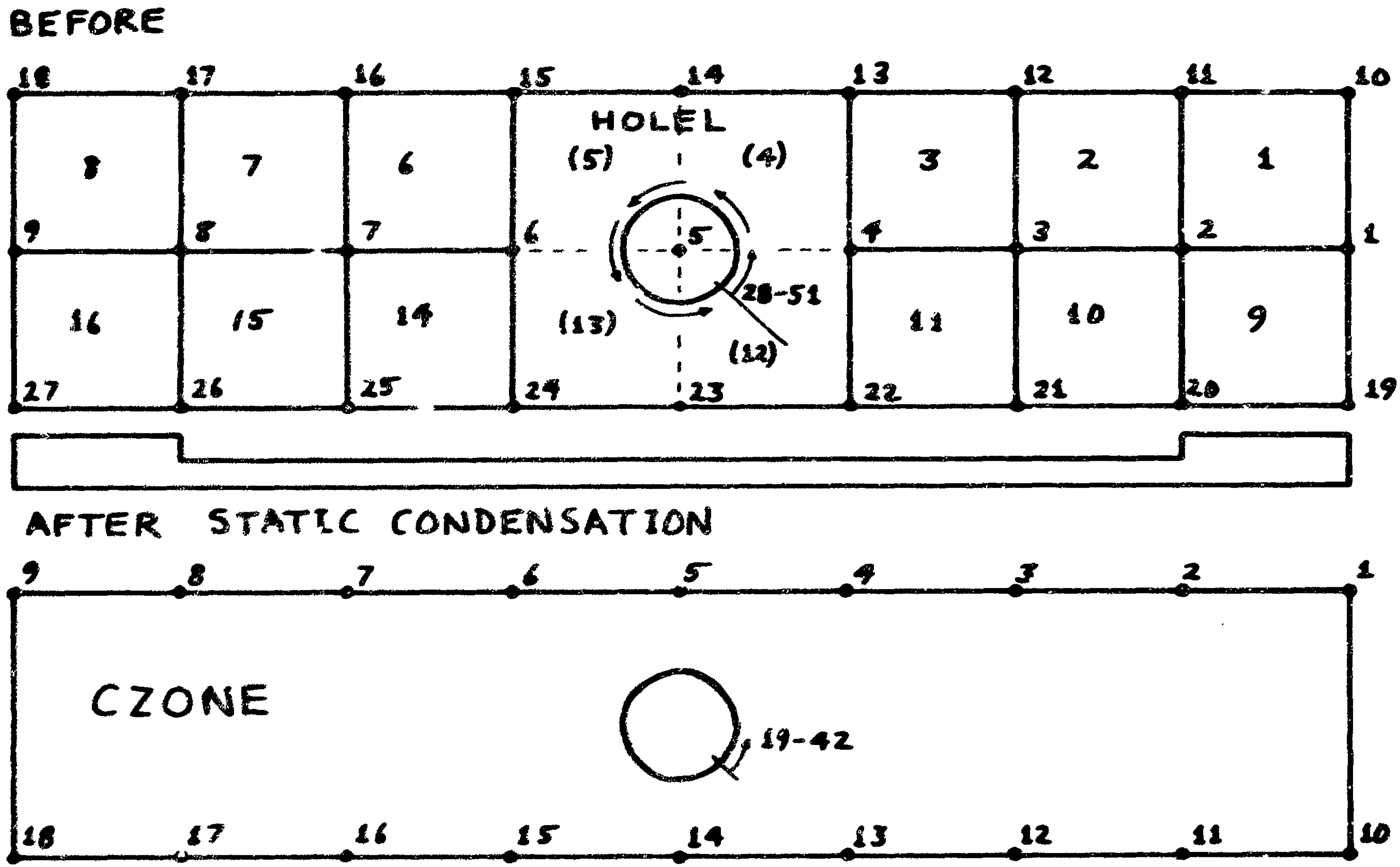
FIG. 23 HIERARCHY OF SUBSTRUCTURED COMPONENTS USED IN PANEL PROGRAM



STATIC CONDENSATION

EXAMPLE WITH 8 ELEMENTS ACROSS PANEL

FIG. 24 LUG SUBSTRUCTURE NUMBERING CONVENTIONS



EXAMPLE WITH 8 ELEMENTS ACROSS PANEL

FIG. 25 CZONE SUBSTRUCTURE NUMBERING CONVENTIONS

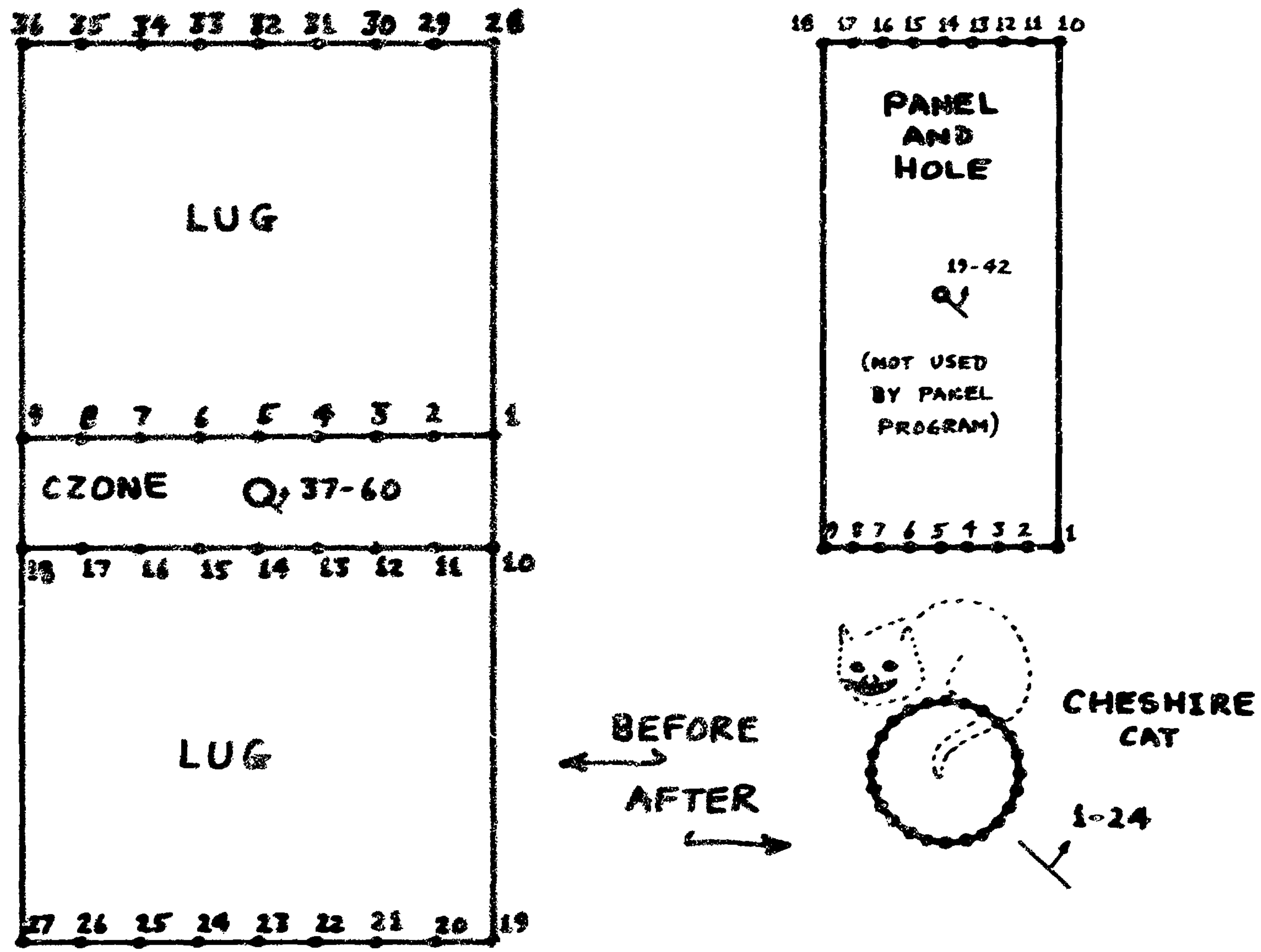


FIG. 26 FARFLD SUBSTRUCTURE NUMBERING CONVENTIONS

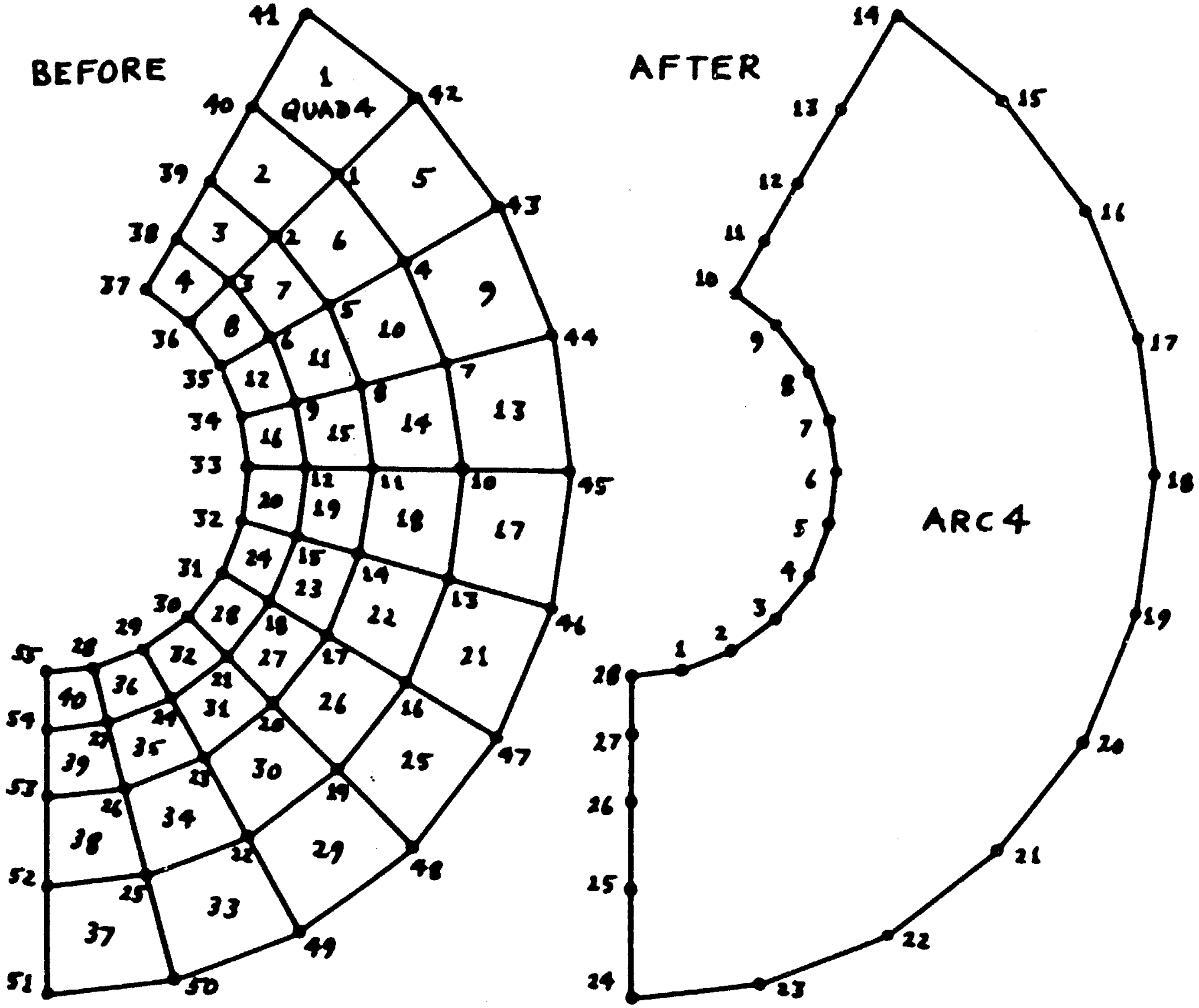
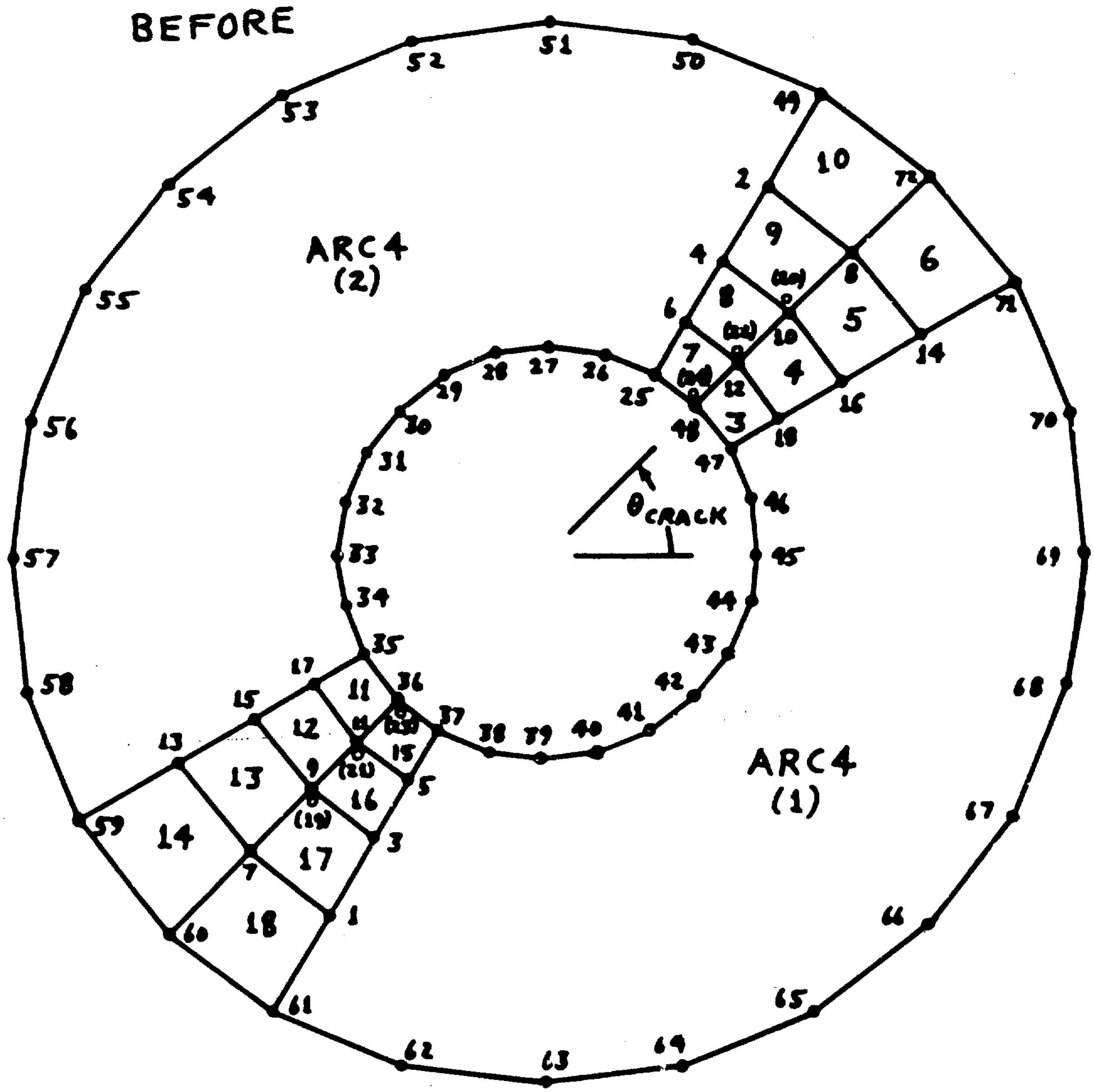
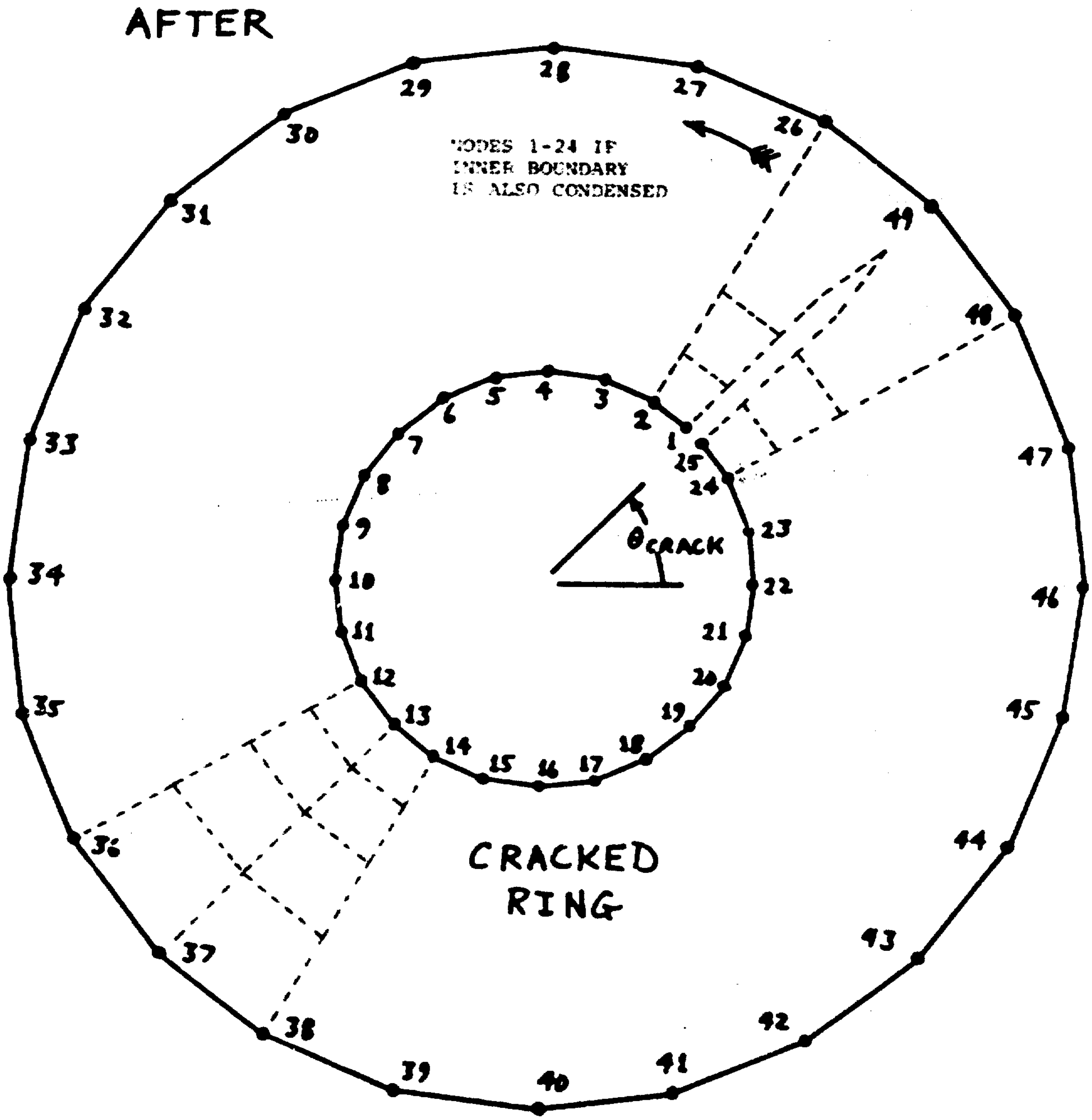


FIG. 27 ARC4 SUBSTRUCTURE NUMBERING CONVENTIONS



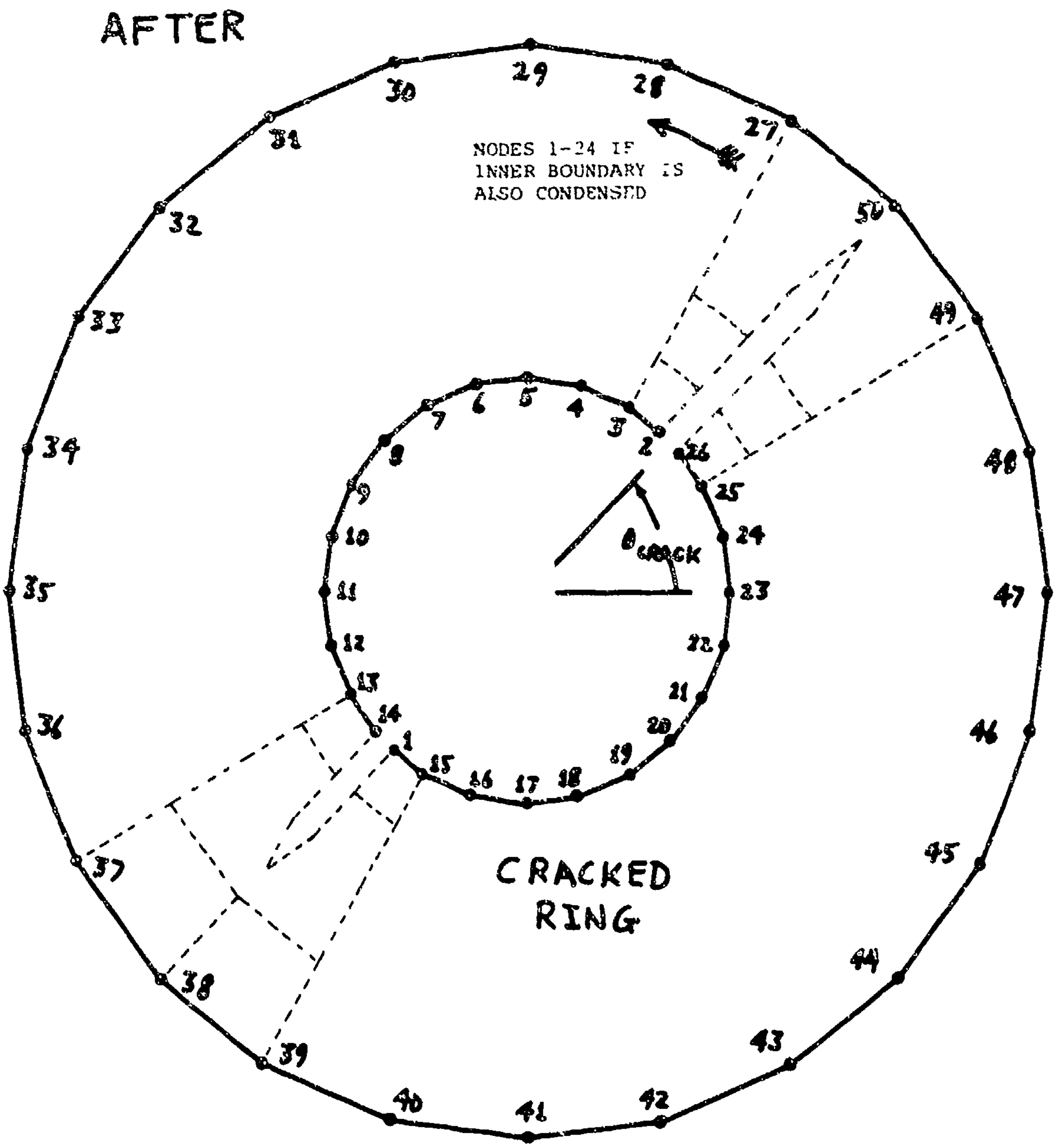
NODES (19)-(24) USED ONLY WHEN CRACK IS PRESENT
PCRK59 ELEMENTS ARE 19 (CRACK 1) AND 20 (CRACK 2)

FIG. 28 RING SUBSTRUCTURE CONVENTIONS (BEFORE CONDENSATION)



EXAMPLE WITH [illegible] 0.8

FIG. 29 RING SUBSTRUCTURE CONVENTIONS (ONE CRACK)



EXAMPLE WITH A(2) ≠ 0.0

FIG. 30 RING SUBSTRUCTURE CONVENTIONS (TWO CRACKS)

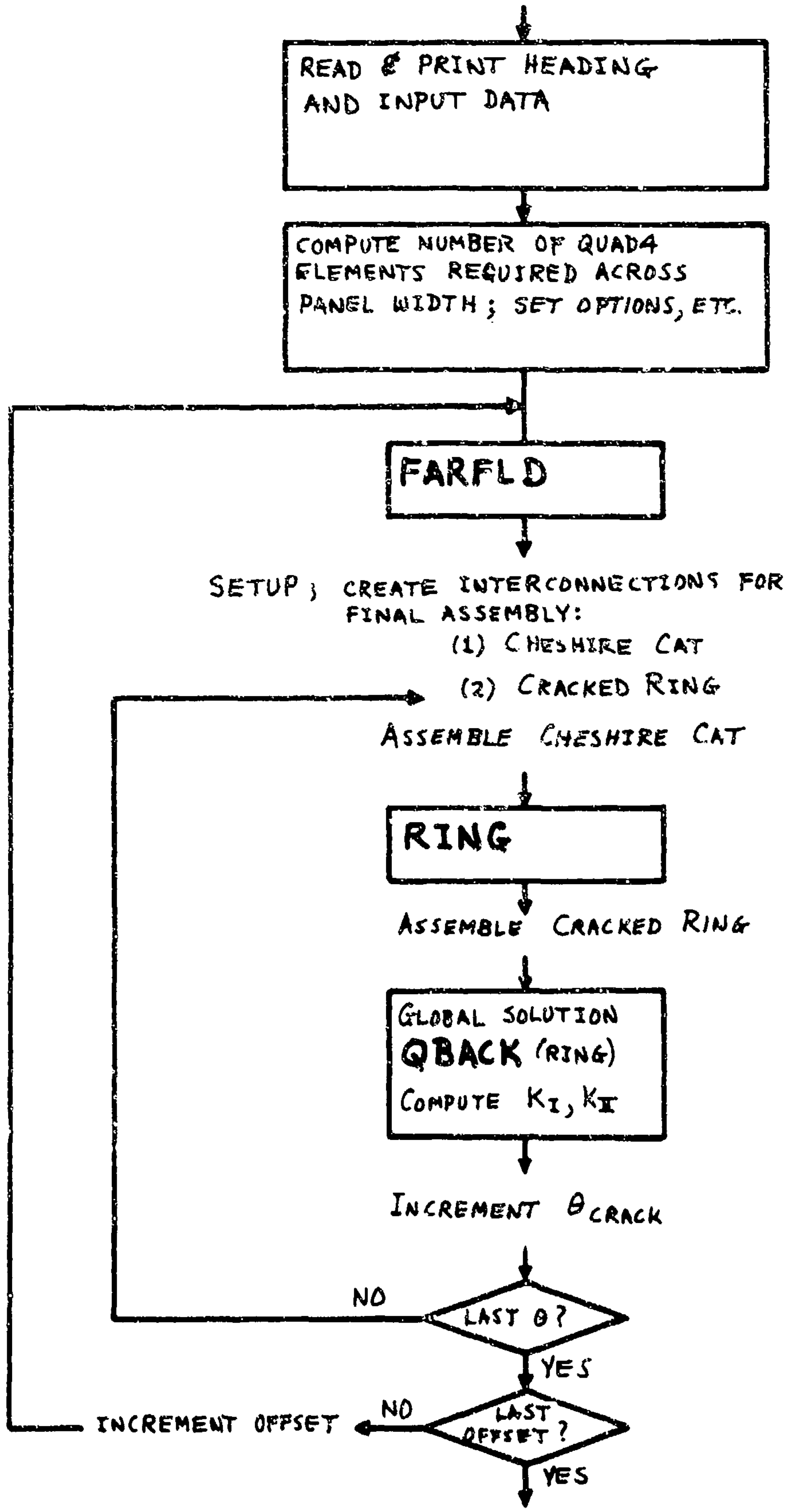


FIG. 31 EXECUTIVE FLOW CHART FOR PANEL PROGRAM

```
                                                  CONTRACT 81739

                                                PROBLEM NO. 3

                                              LEFT SIDE STIFFENED

                                                   INPUT DATA

PLATE GEOMETRY:

PLATE WIDTH= 0.40000E+01
PLATE LENGTH= 0.10000E+02
PLATE THICKNESS= 0.10000E+01
PLATE STIFFENER FACTOR=   0.500
FAR FIELD LOADING (PSI)= 0.10000E+04
RADIUS OF HOLE= 0.12500E+00
HOLE OFFSET INDICATOR=  1(=0, HOLE REMAINS CENTERED
                          =-1, HOLE MOVED TO THE RIGHT
                          =+1, HOLE MOVED TO THE LEFT)

CRACK DATA:
CRACK NO.   1 :  CRACK LENGTH= 0.12500E+00
                 INITIAL CRACK POSITION=  1(   0.0    DEGREES)
                   FINAL CRACK POSITION= 13( 180.000 DEGREES)

CRACK NO.   2 :  CRACK LENGTH= 0.12500E+00
                 INITIAL CRACK POSITION=  1( 180.000 DEGREES)
                   FINAL CRACK POSITION= 13( 360.000 DEGREES)


MATERIAL PROPERTIES:

YOUNGS MODULUS (E)= 0.10000E+08
POISSONS RATIO= 0.30000E+00




CRACK NO. 1  ANGLE=    0.0      KI= 0.87369E+03    KII= 0.11424E+00
CRACK NO. 2  ANGLE= 180.000     KI= 0.87216E+03    KII= 0.45826E+00


CRACK NO. 1  ANGLE=  15.000     KI= 0.87397E+03    KII= 0.35178E+03
CRACK NO. 2  ANGLE= 195.000     KI= 0.87177E+03    KII= 0.35054E+03


CRACK NO. 1  ANGLE=  30.000     KI= 0.82898E+03    KII= 0.58763E+03
CRACK NO. 2  ANGLE= 210.000     KI= 0.82647E+03    KII= 0.58543E+03


CRACK NO. 1  ANGLE=  45.000     KI= 0.60824E+03    KII= 0.58508E+03
CRACK NO. 2  ANGLE= 225.000     KI= 0.60659E+03    KII= 0.58246E+03


CRACK NO. 1  ANGLE=  60.000     KI= 0.27609E+03    KII= 0.44344E+03
CRACK NO. 2  ANGLE= 240.000     KI= 0.27596E+03    KII= 0.44213E+03
```

FIG. 32 SAMPLE OUTPUT FROM PANEL PROGRAM

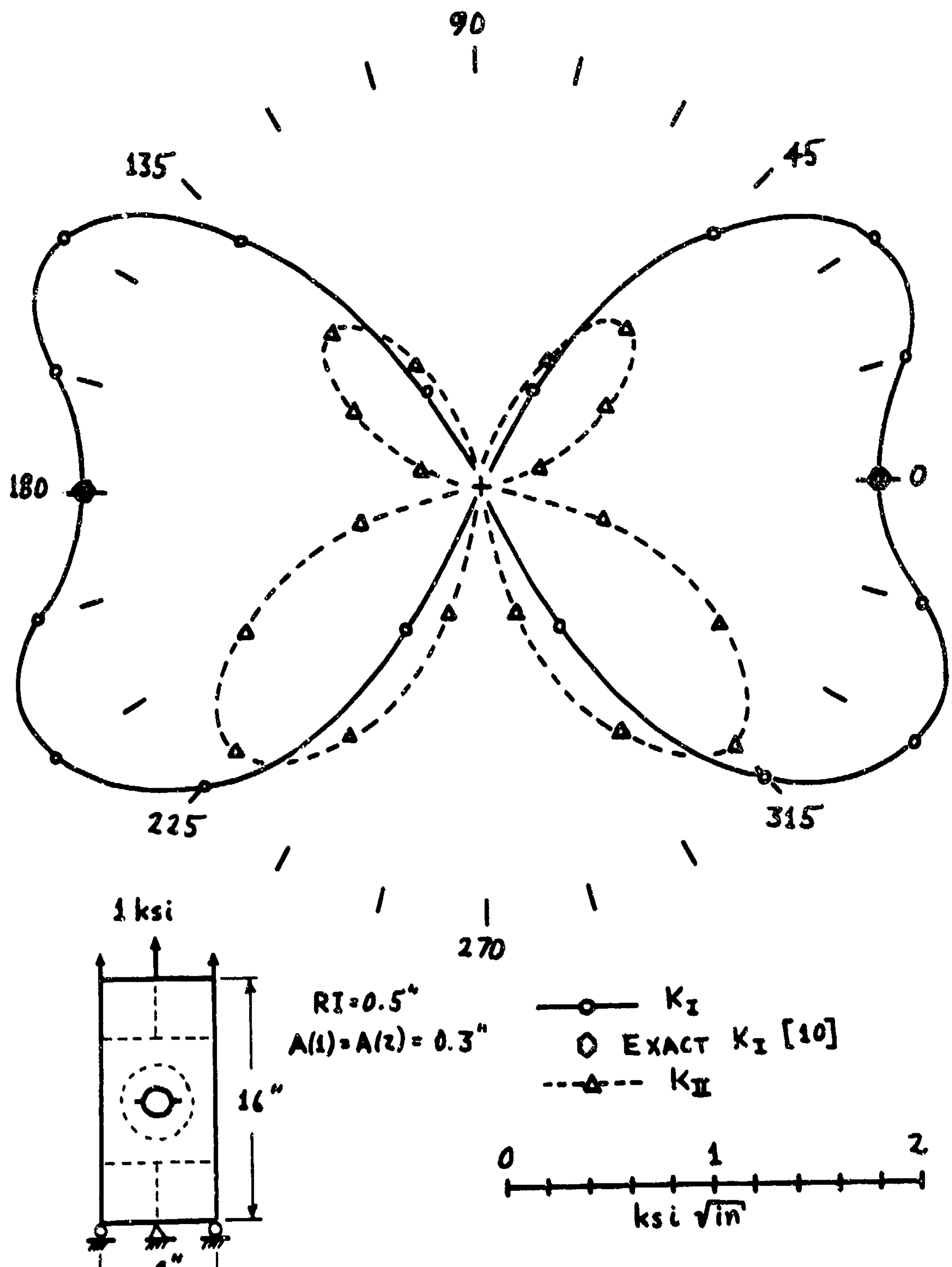


FIG. 33 PANEL BUTTERFLY PLOT FOR $W/D_O=1.59$

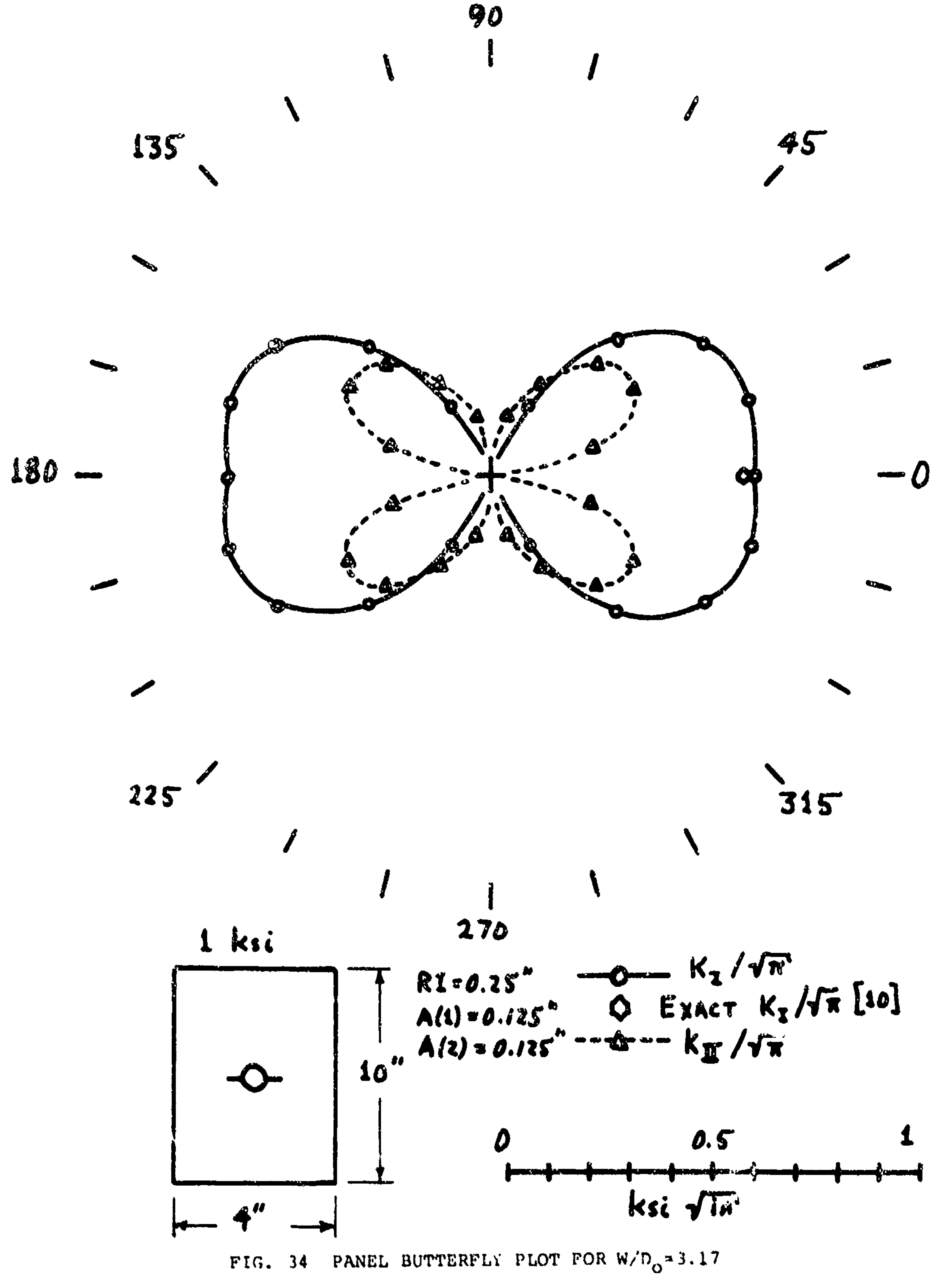


FIG. 34 PANEL BUTTERFLY PLOT FOR $W/D_0 = 3.17$

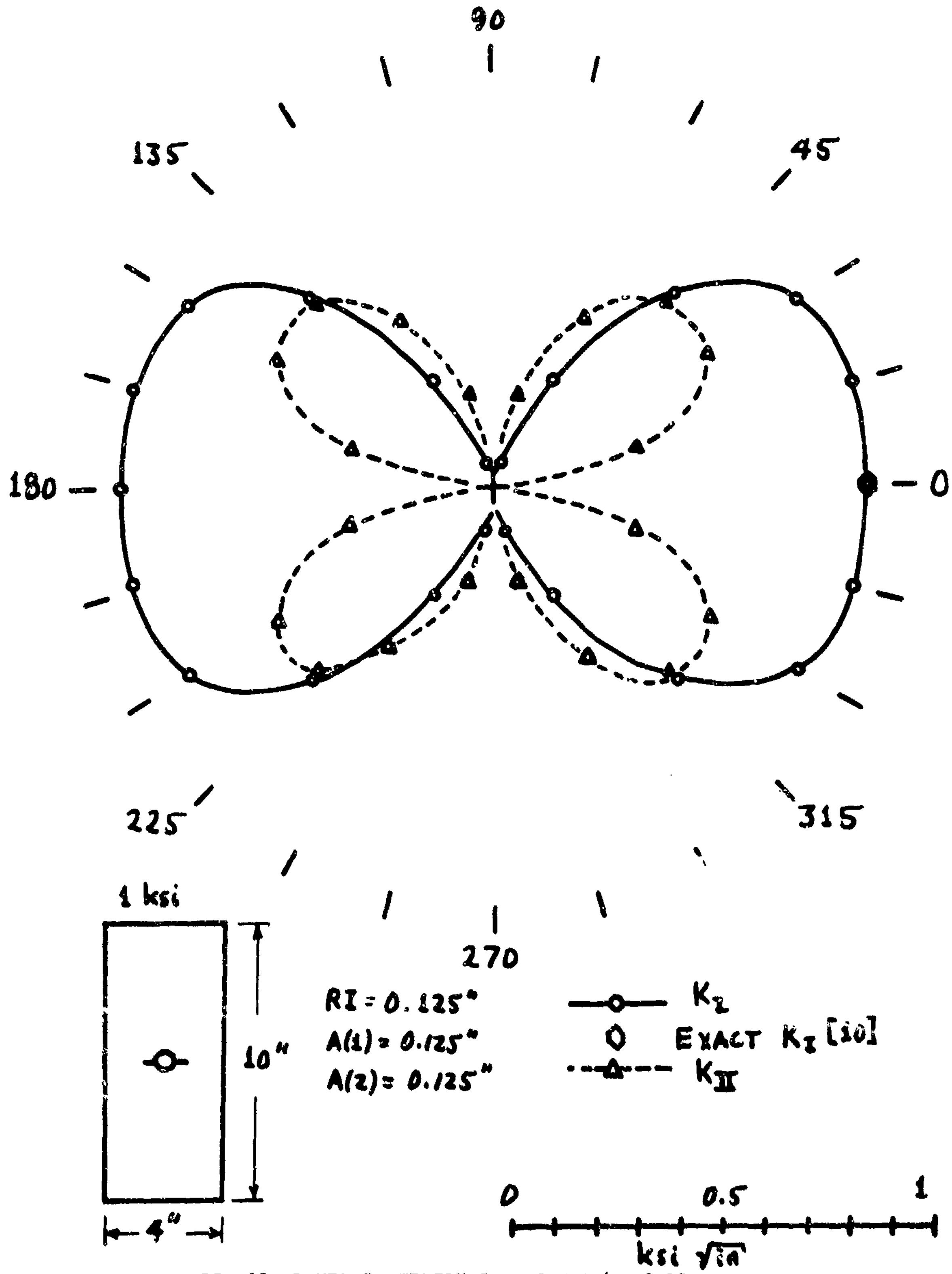


FIG. 35 PANEL BUTTERFLY PLOT FOR W/D_O=6.35

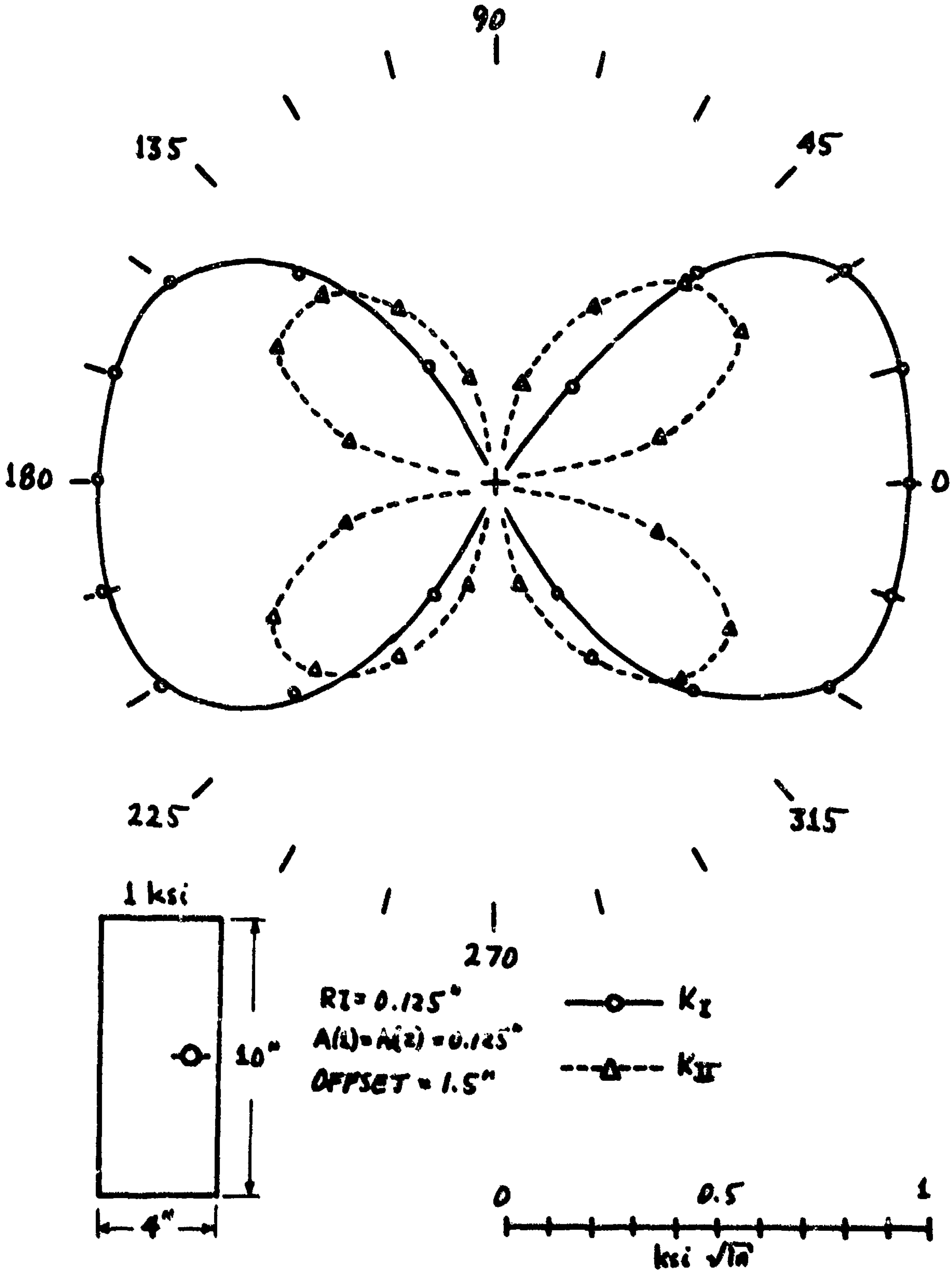


FIG. 36 BUTTERFLY PLOT FOR PANEL WITH W/D_o=6.35 AND FASTENER HOLE OFFSET 1.5 INCHES TO RIGHT

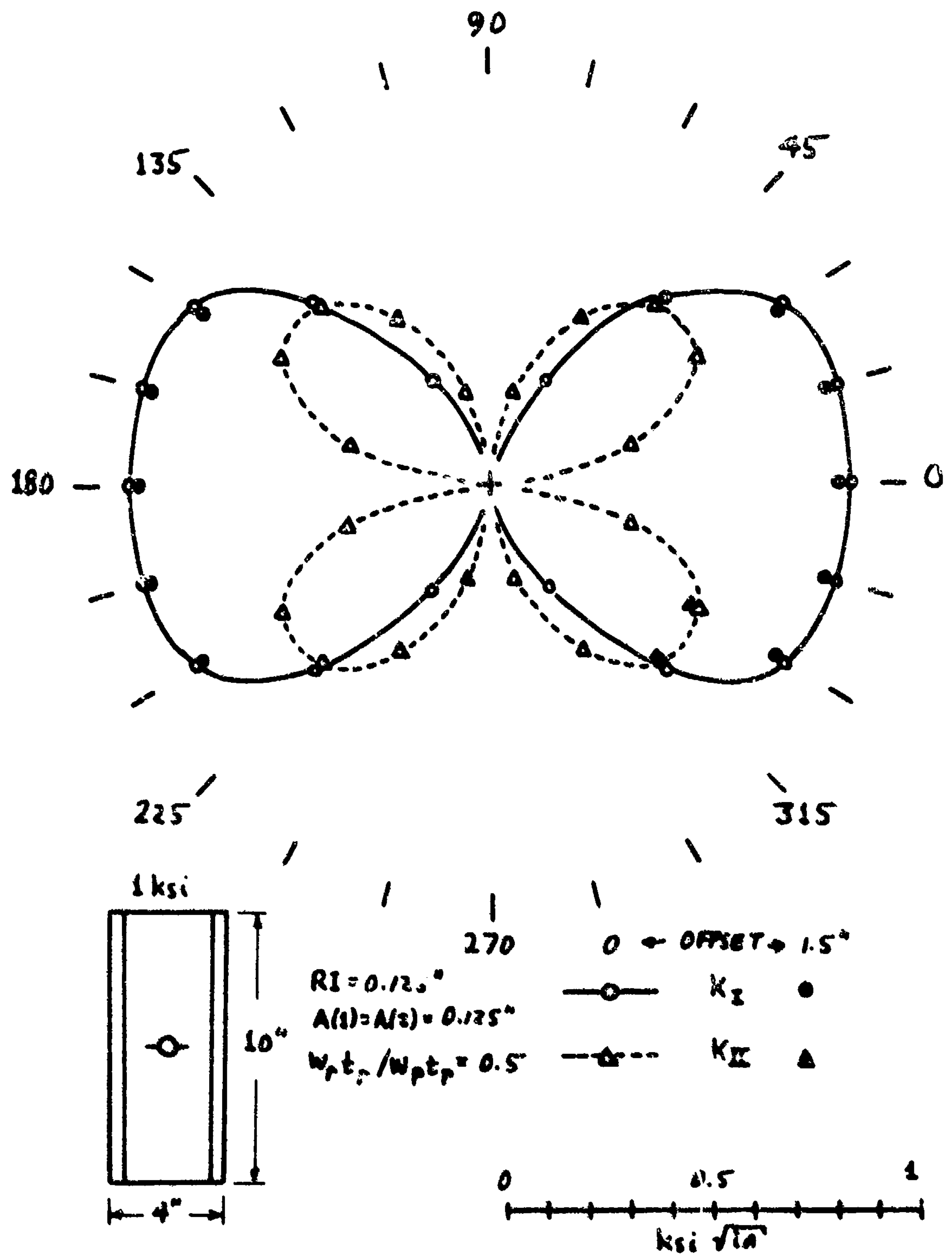


FIG. [illegible] BUTTERFLY PLOT FOR SYMMETRICALLY STIFFENED PANEL WITH STIFFENER FACTOR = 0.5

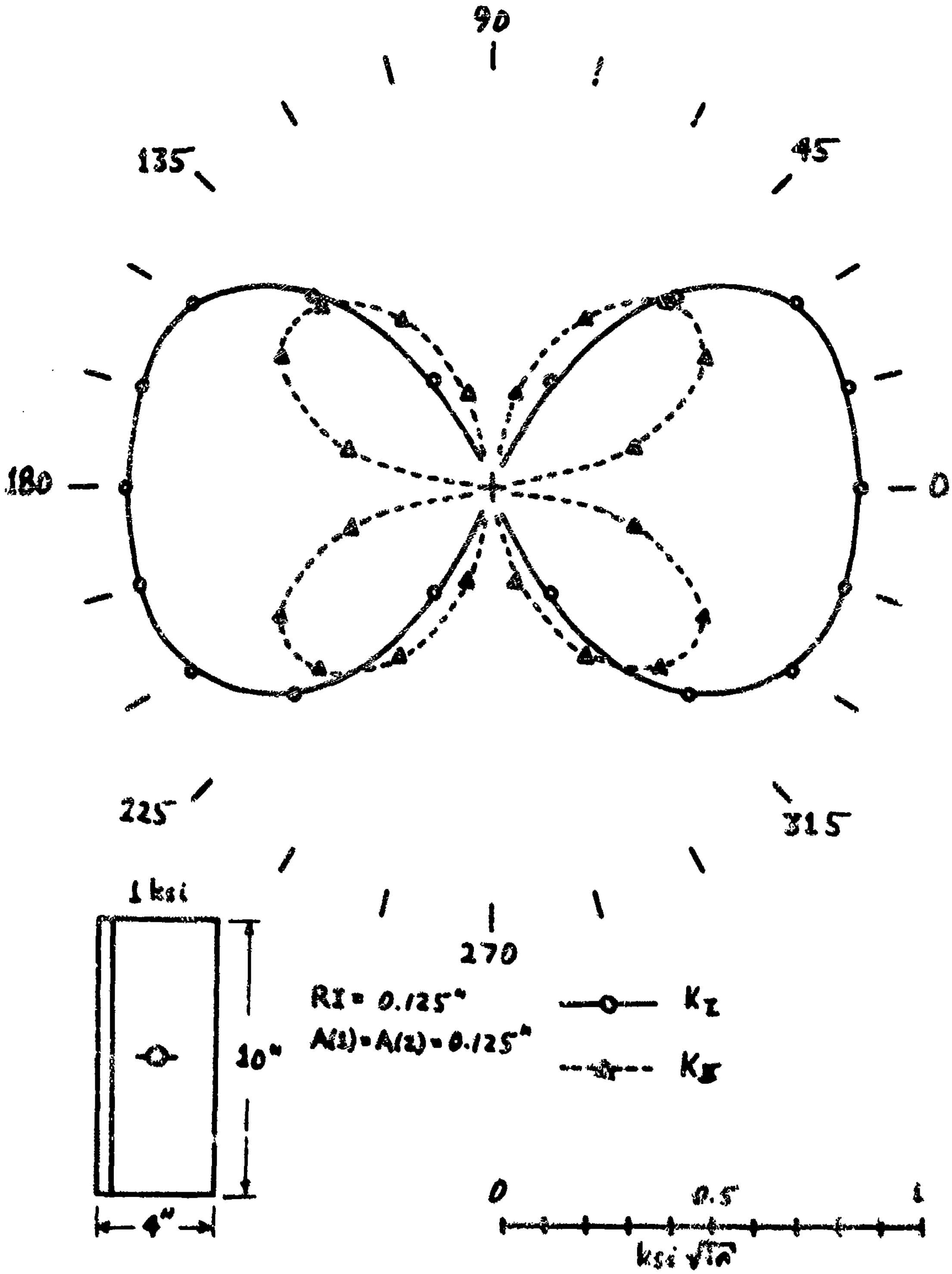


FIG. 38 BUTTERFLY PLOT FOR PANEL WITH LEFT EDGE STIFFENED (FACTOR = 0.5) AND HOLE CENTERED

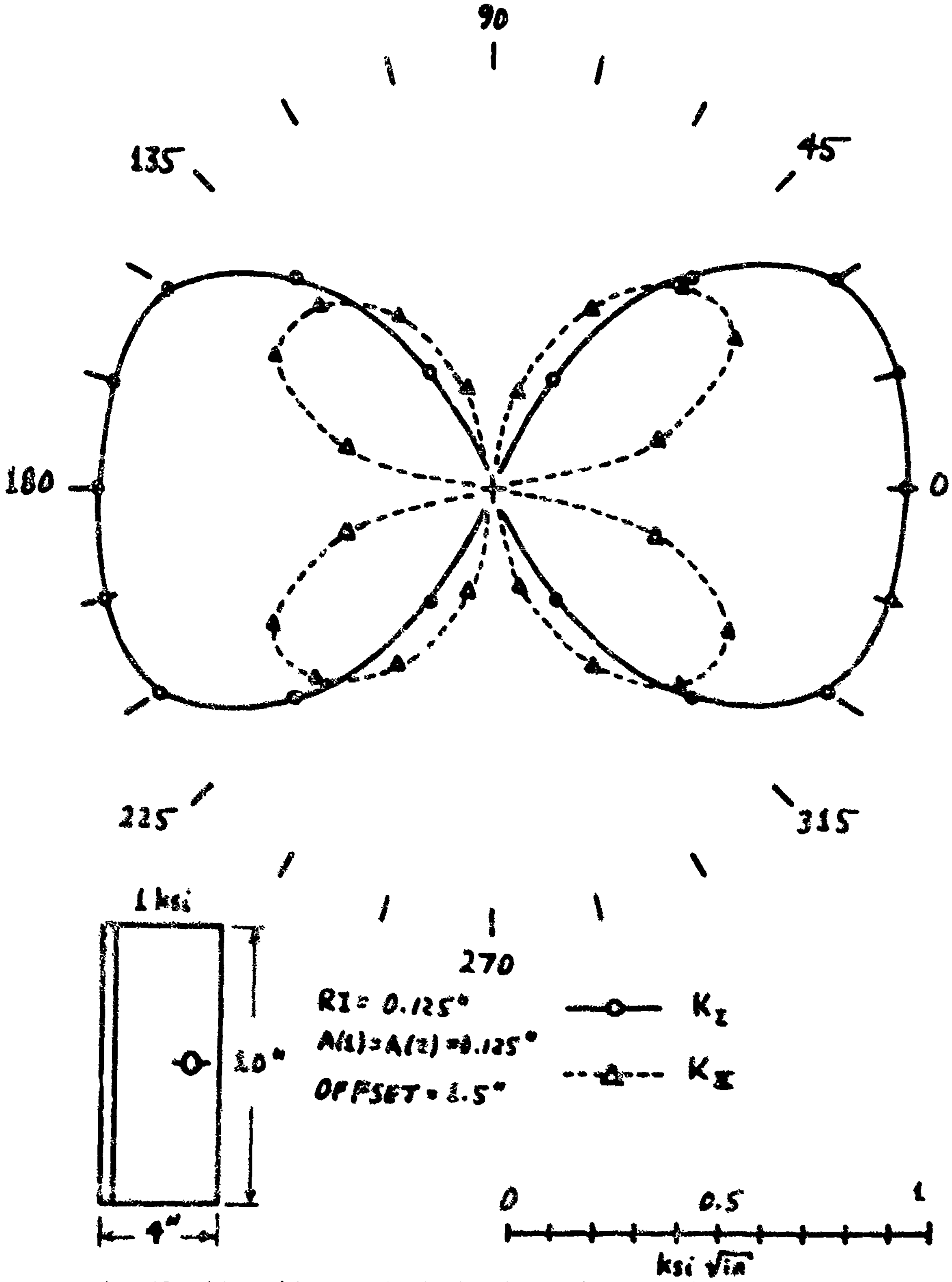


FIG. 39 BUTTERFLY PLOT FOR PANEL WITH LEFT EDGE STIFFENED AND HOLE OFFSET 1.5 INCHES TO RIGHT

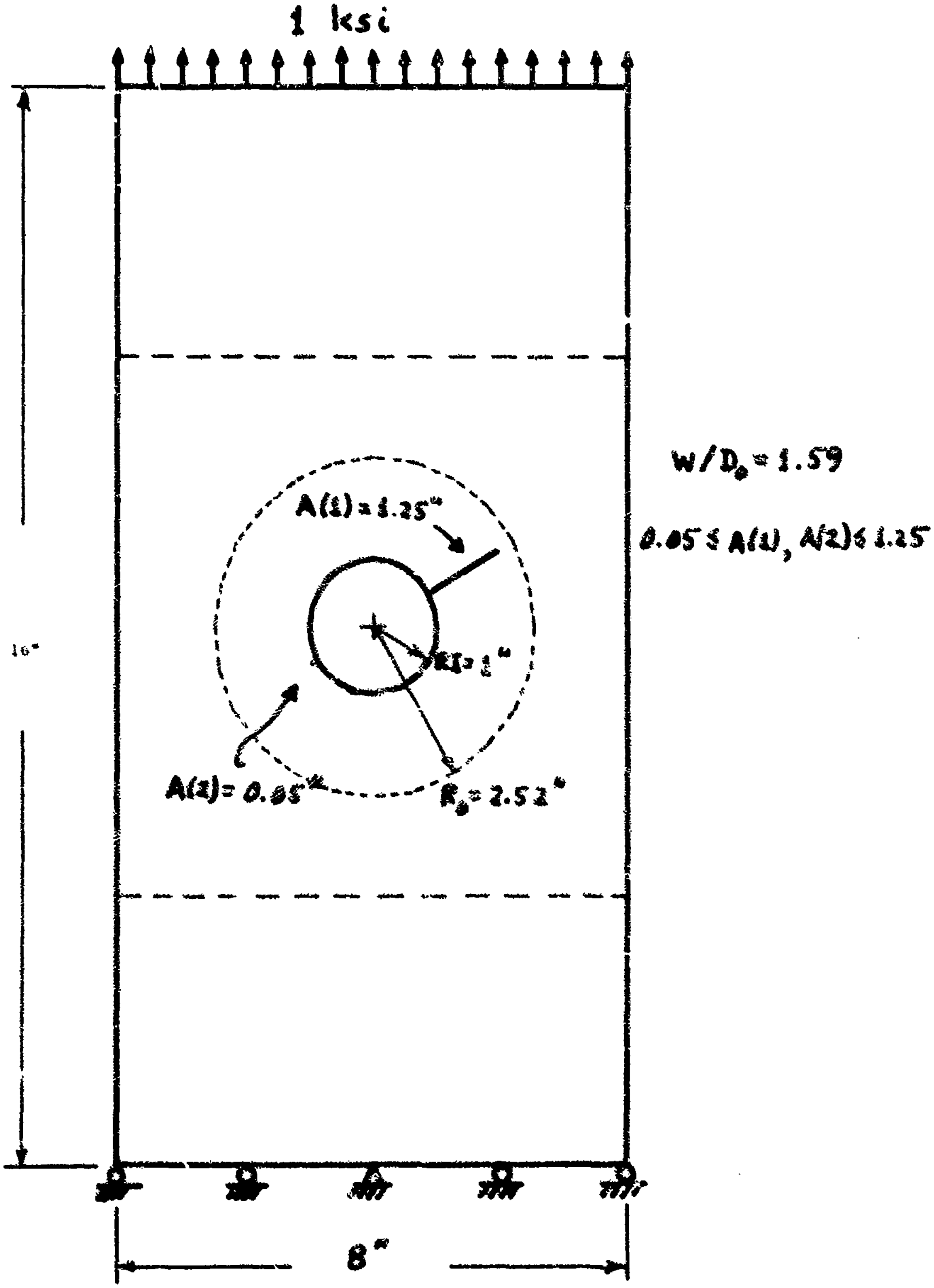


FIG. 40 PANEL DIMENSIONS FOR TEST OF SENSITIVITY TO a/R_1

Appendix A

ALGORITHM FOR TRANSFERRING A SUBSTRUCTURE TO RESERVE STORAGE

Assume that a standard FEABL-2 assembly has been executed, and that the assembled structure has been statically condensed with subroutine STACON. The objective is to transfer the condensed stiffness matrix $K_{BB}^{(c)}$ and force $\hat{Q}_B^{(c)}$ from the FEABL DATA vector (designated by RNAME,INAME) to reserve storage. Assume further that $K_{BB}^{(c)}$ is almost fully populated. Therefore, $K_{BB}^{(c)}$ is to be inflated to a fully populated, lower-triangle-vector-stored matrix during the transfer process. For this purpose, vectors STORK and STORQ are dimensioned in the MAIN program to at least N(N+1)/2 and N words respectively, where N is the total number of boundary (uncondensed) degrees of freedom remaining. Vector STORK will receive $K_{BB}^{(c)}$, while STORQ receives $\hat{Q}_B^{(c)}$.

Other variables appearing in the algorithm play the following roles. IZERO is the first uncondensed degree of freedom, i.e., IZERO = NIO+1. I and J are loop indices which control progress through $K_{BB}^{(c)}$ and $\hat{Q}_B^{(c)}$ in the DATA vector: I for the rows and J for the columns. If a row is encountered for which the band margin of $K_{BB}^{(c)}$ lies to the right of column IZERO, then some leading zeros must be inserted in the corresponding row in STORK. LNZCOL is used to compute and compare the band margin. II is used to hold the variable-bandwidth address information required to locate

stiffnesses in the DATA vector belonging to ROW I. M, N and MN are used to trace the lower-triangle address in vector STORK. IKOUNT, ILNZ and IQ are FEABL address control parameters located in the BEGIN labelled COMMON area. NDT is the total number of degrees of freedom in the assembled structure (boundary plus interior), available in the SIZE labelled COMMON area.

The transfer algorithm begins immediately after subroutine STACON has been executed:

```
          .
          .
          .
          IZERO = NID+1
          M=0
          DO 20 I=IZERO, NDT
          II=INAME(IKOUNT+I-1)
          M=M+1
          STORQ(M)=RNAME(IQ+I-1)
          N=0
          DO 10 J=IZERO,I
          N=N+1
          MN=(M*(M-1))/2+N
          STORK(MN)=0.0
          LNZCOL=INAME(ILNZ+I-1)
          IF(LNZCOL-J) 5,5,10
    5     STORK(MN) = RNAME(II+J)
   10     CONTINUE
   20     CONTINUE
          .
          .
          .
```

APPENDIX B

```
      SUBROUTINE HOLEL(COORD,THK,S,RI,RSS,ISS,B)                       HOLE0000
C**********************************************************************HOLE0001
C SUBROUTINE HOLEL GENERATES A RECTANGULAR FINITE ELEMENT SUBSTRUCTURE  HOLE0002
C WITH A ROUND HOLE IN ITS CENTER. THERE ARE 8 NODES ON ITS PERIMETER   HOLE0003
C AND 24 NODES AROUND THE HOLE                                          HOLE0004
C**********************************************************************HOLE0005
C COPYRIGHT (C) 1975 MASSACHUSETTS INSTITUTE OF TECHNOLOGY              HOLE0006
C AEROELASTIC AND STRUCTURES RESEARCH LABORATORY                        HOLE0007
C**********************************************************************HOLE0008
C FINITE ELEMENT APPLICATIONS TO USAF STRUCTURAL INTEGRITY PROBLEMS     HOLE0009
      DIMENSION RSS(2097),ISS(2097),ISAVE1(2),ISAVE2(6),ISAVE3(6)      HOLE0010
     @,COORD(12),S(3,3),V(78),ELQ(12),B(6,3,13),VR(78),T(12,12),       HOLE0011
     @ELKR(12,12),ELK(12,12),NODEL(6,8)                                HOLE0012
      COMMON/IO/KR,KW,KP,KT1,KT2,KT3                                   HOLE0013
      COMMON/SIZE/NET,NDT                                              HOLE0014
      COMMON/BEGIN/IBEGIN(6)                                           HOLE0015
      COMMON/END/IEND(6)                                               HOLE0016
      COMMON/SIZESS/NETSS,NDTSS,NIDSS                                  HOLE0017
      COMMON/BEGSS/IBG(6)                                              HOLE0018
      DATA PI/3.141593/                                                HOLE0019
      DATA NODEL/2,1,9,10,11,12,2,3,15,14,13,12,                       HOLE0020
     @4,3,15,16,17,18,4,5,21,20,19,18,                                 HOLE0021
     @6,5,21,22,23,24,6,7,27,26,25,24,                                 HOLE0022
     @8,7,27,28,29,30,8,1,9,32,31,30/                                  HOLE0023
      DO 130 I=1,12                                                    HOLE0024
      DO 130 J=1,I                                                     HOLE0025
      T(I,J)=0.0                                                       HOLE0026
  130 T(J,I)=0.0                                                       HOLE0027
C  SAVE CALLING PROGRAM FEABL CONTROLS                                  HOLE0028
      ISAVE1(1)=NET                                                    HOLE0029
      ISAVE1(2)=NDT                                                    HOLE0030
      DO 10 I=1,6                                                      HOLE0031
      ISAVE2(I)=IBEGIN(I)                                              HOLE0032
   10 ISAVE3(I)=IEND(I)                                                HOLE0033
      PI4=PI/4.0                                                       HOLE0034
      PI2=PI/2.0                                                       HOLE0035
```

```
      NET=8                                                             HOLE0036
      NDT=64                                                            HOLE0037
      CALL SETUP(2097,9,104,RSS,ISS)                                    HOLE0038
C  ESTABLISH ASSEMBLY LIST                                              HOLE0039
      IMASTR=IBEGIN(4)                                                  HOLE0040
      LMASTR=IEND(4)                                                    HOLE0041
      DO 40 N=1,NET                                                     HOLE0042
      NM1=N-1                                                           HOLE0043
      IPNTR=IMASTR+NET+NM1*12                                           HOLE0044
      ISS(IMASTR+NM1)=IPNTR                                             HOLE0045
      DO 40 I=1,6                                                       HOLE0046
      IADD=IPNTR+I*2-1                                                  HOLE0047
      IDOF=NODEL(I,N)*2                                                 HOLE0048
      ISS(IADD-1)=IDOF-1                                                HOLE0049
   40 ISS(IADD)=IDOF                                                    HOLE0050
      IF(KT1 .EQ. KW) GO TO 707                                         HOLE0051
      WRITE(KW,6001)(ISS(I),I=IMASTR,LMASTR)                            HOLE0052
 6001 FORMAT(30H MASTER ASSEMBLY LIST:  HOLEL:,/,10H POINTERS:,8I5,/,16HHOLE0053
     @ ELEMENT D.O.F.S.,/,8(1H ,12I5,/))                                HOLE0054
  707 CONTINUE                                                          HOLE0055
      CALL ORK(2097,RSS,ISS)                                            HOLE0056
C  DETERMINE NODE/QUADRANT ASSOC.                                       HOLE0057
      XC=0.0                                                            HOLE0058
      YC=0.0                                                            HOLE0059
      DO 50 I=1,4                                                       HOLE0060
      N=I*3-3                                                           HOLE0061
      XC=XC+COORD(N+1)/4.0                                              HOLE0062
   50 YC=YC+COORD(N+2)/4.0                                              HOLE0063
      DO 60 I=1,4                                                       HOLE0064
      N=I*3-3                                                           HOLE0065
      COORD(N+1)=COORD(N+1)-XC                                          HOLE0066
   60 COORD(N+2)=COORD(N+2)-YC                                          HOLE0067
      THETAI=0.0                                                        HOLE0068
      IF(COORD(1) .GT. 0.0 .AND. COORD(2) .GT. 0.0) THETAI=PI2          HOLE0069
      IF(COORD(1) .LT. 0.0 .AND. COORD(2) .GT. 0.0) THETAI=PI           HOLE0070
      IF(COORD(1) .LT. 0.0 .AND. COORD(2) .LT. 0.0) THETAI=PI+PI2       HOLE0071
```

```
      HWD=ABS(COORD(1))                                               HOLE0072
      CALL QHOLEL(THK,S,PI4,HWD,RI,ELK,ELQ,H)                         HOLE0073
C     OBTAIN REFLECTION                                               HOLE0074
      T(1,1)=1.0                                                      HOLE0075
      T(2,2)=-1.0                                                     HOLE0076
      DO 70 IT=2,12,2                                                 HOLE0077
      ITM2=IT-2                                                       HOLE0078
      DO 70 I=1,2                                                     HOLE0079
   70 T(ITM2+I,ITM2+I)=T(I,I)                                         HOLE0080
      DO 80 I=1,12                                                    HOLE0081
      DO 80 J=1,12                                                    HOLE0082
      ELKR(I,J)=0.0                                                   HOLE0083
      DO 80 K=1,12                                                    HOLE0084
      DO 80 L=1,12                                                    HOLE0085
   80 ELKR(I,J)=ELKR(I,J)+T(K,I)*ELK(K,L)*T(L,J)                      HOLE0086
C     TRANSFORM ELEMENTS                                              HOLE0087
      THETA =THETAI                                                   HOLE0088
      NEL1=-1                                                         HOLE0089
      NEL2=0                                                          HOLE0090
      DO 110 N=1,4                                                    HOLE0091
      NEL1=NEL1+2                                                     HOLE0092
      NEL2=NEL2+2                                                     HOLE0093
      GO TO (1,2,3,4),N                                               HOLE0094
    1 CS=1.0                                                          HOLE0095
      SS=0.0                                                          HOLE0096
      GO TO 5                                                         HOLE0097
    2 CS=0.0                                                          HOLE0098
      SS=1.0                                                          HOLE0099
      GO TO 5                                                         HOLE0100
    3 CS=-1.0                                                         HOLE0101
      SS=0.0                                                          HOLE0102
      GO TO 5                                                         HOLE0103
    4 CS=0.0                                                          HOLE0104
      SS=-1.0                                                         HOLE0105
    5 CONTINUE                                                        HOLE0106
      T(1,1)=CS                                                       HOLE0107
```

```
      T(1,2)=SS                                                         HOLE0108
      T(2,1)=-T(1,2)                                                    HOLE0109
      T(2,2)=T(1,1)                                                     HOLE0110
      DO 90 IT=2,12,2                                                   HOLE0111
      ITM2=IT-2                                                         HOLE0112
      DO 90 I=1,2                                                       HOLE0113
      DO 90 J=1,2                                                       HOLE0114
   90 T(ITM2+I,ITM2+J)=T(I,J)                                           HOLE0115
      L=0                                                               HOLE0116
      DO 100 I=1,12                                                     HOLE0117
      DO 100 J=1,I                                                      HOLE0118
      L=L+1                                                             HOLE0119
      V(L)=0.0                                                          HOLE0120
      VR(L)=0.0                                                         HOLE0121
      DO 100 K=1,12                                                     HOLE0122
      DO 100 LL=1,12                                                    HOLE0123
      V(L)=V(L)+T(K,I)*ELK(K,LL)*T(LL,J)                                HOLE0124
  100 VR(L)=VR(L)+T(K,I)*ELKR(K,LL)*T(LL,J)                             HOLE0125
      CALL ASMLTV(NEL1,12,VR,FLQ,RSS,ISS)                               HOLE0126
      CALL ASMLTV(NEL2,12,V,FLQ,RSS,ISS)                                HOLE0127
  110 THETA=THETA+PI2                                                   HOLE0128
C  RETURN CALLING PROGRAM FEARL CONTROLS TO THEIR ORIGINAL STORAGE      HOLE0129
      NETSS=NET                                                         HOLE0130
      NDTSS=NDT                                                         HOLE0131
      NIDSS=0                                                           HOLE0132
      NET=ISAVE1(1)                                                     HOLE0133
      NDT=ISAVE1(2)                                                     HOLE0134
      DO 120 I=1,6                                                      HOLE0135
      IBG(I)=IBEGIN(I)                                                  HOLE0136
      IBEGIN(I)=ISAVE2(I)                                               HOLE0137
  120 IEND(I)=ISAVE3(I)                                                 HOLE0138
      RETURN                                                            HOLE0139
      END                                                               HOLE0140
```

```
      SUBROUTINE QHOLEL(THK,S,PI4,HWD,RI,ELK,ELQ,B)                     QHOL0000
C***********************************************************************QHOL0001
C SUBROUTINE QHOLEL FORMS STIFFNESS MATRIX AND B MATRIX                 QHOL0002
C THIS SUBROUTINE IS USED IN FORMING THE HOLE SUBSTRUCTURE OF HOLEL     QHOL0003
C***********************************************************************QHOL0004
C COPYRIGHT (C) 1975 MASSACHUSETTS INSTITUTE OF TECHNOLOGY              QHOL0005
C AEROELASTIC AND STRUCTURES RESEARCH LABORATORY                        QHOL0006
C***********************************************************************QHOL0007
C FINITE ELEMENT APPLICATIONS TO USAF STRUCTURAL INTEGRITY PROBLEMS     QHOL0008
      DIMENSION S(3,3),ELK(12,12),HI(10,10),B(6,3,13),G(10,12),HIG(10,12QHOL0009
     @),P(3,10),ELQ(12)                                                 QHOL0010
      COMMON/TRGFNT/CS,SS,CS2,SS2,CS4,SS4                               QHOL0011
      CALL HMTRX(PI4,HWD,RI,S,HI)                                       QHOL0012
      CALL GMTRX(PI4,HWD,RI,G)                                          QHOL0013
      DO 10 I=1,10                                                      QHOL0014
      DO 10 J=1,12                                                      QHOL0015
      HIG(I,J)=0.0                                                      QHOL0016
      DO 10 K=1,10                                                      QHOL0017
   10 HIG(I,J)=HIG(I,J)+HI(I,K)*G(K,J)                                  QHOL0018
      DO 20 I=1,12                                                      QHOL0019
      ELQ(I)=0.0                                                        QHOL0020
      DO 20 J=1,12                                                      QHOL0021
      ELK(I,J)=0.0                                                      QHOL0022
      DO 20 K=1,10                                                      QHOL0023
   20 ELK(I,J)=ELK(I,J)+G(K,I)*HIG(K,J)*THK                             QHOL0024
      R=RI                                                              QHOL0025
      THETA=PI4/2.0                                                     QHOL0026
      CALL TRIG(THETA)                                                  QHOL0027
      DR=(HWD/CS-RI)/5.0                                                QHOL0028
      DO 40 I=1,6                                                       QHOL0029
      CALL PMTRX(P,R)                                                   QHOL0030
      DO 30 J=1,3                                                       QHOL0031
      DO 30 K=1,12                                                      QHOL0032
      B(I,J,K)=0.0                                                      QHOL0033
      DO 30 L=1,10                                                      QHOL0034
   30 B(I,J,K)=B(I,J,K)+P(J,L)*HIG(L,K)                                 QHOL0035
```

```
      B(I,1,13)=THETA                                                   QHOL0036
      B(I,2,13)=R                                                       QHOL0037
      B(I,3,13)=0.0                                                     QHOL0038
   40 R=R+DR                                                            QHOL0039
      RETURN                                                            QHOL0040
      END                                                               QHOL0041
```

```
      SUBROUTINE HMTRX(PI4,HWD,RI,S,H)                                  HMTR0000
C***********************************************************************HMTR0001
C SUBROUTINE HMTRX GENERATES MATRICES H AND H INVERSE                   HMTR0002
C THIS SUBROUTINE IS USED IN FORMING THE HOLE SUBSTRUCTURE OF HOLEL     HMTR0003
C***********************************************************************HMTR0004
C COPYRIGHT (C) 1975 MASSACHUSETTS INSTITUTE OF TECHNOLOGY              HMTR0005
C AEROELASTIC AND STRUCTURES RESEARCH LABORATORY                        HMTR0006
C***********************************************************************HMTR0007
C FINITE ELEMENT APPLICATIONS TO USAF STRUCTURAL INTEGRITY PROBLEMS     HMTR0008
      DIMENSION            H(10,10),HV(55),S(3,3),P(3,10)               HMTR0009
      DIMENSION Z(5),WZ(5)                                              HMTR0010
      COMMON/TRGFNT/CS,SS,CS2,SS2,CS4,SS4                               HMTR0011
      F(A,B,Z)=A*Z+B                                                    HMTR0012
      DATA WZ/.2369269        ,.4786287        ,.5688889                HMTR0013
     &.4786287        ,.2369269        /                                HMTR0014
      DATA Z/-.9061798        ,-.5384693        ,0.0E0,                 HMTR0015
     &.5384693        ,.9061798        /,NZ/5/                          HMTR0016
      DO 10 I=1,10                                                      HMTR0017
      DO 10 J=1,I                                                       HMTR0018
      H(I,J)=0.0                                                        HMTR0019
   10 H(J,I)=H(I,J)                                                     HMTR0020
      DELTHT=PI4/2.0                                                    HMTR0021
      AVGTHT=DELTHT                                                     HMTR0022
      DO 20 ITHETA=1,NZ                                                 HMTR0023
      CALL TRIG(F(DELTHT,AVGTHT,Z(ITHETA)))                             HMTR0024
      RO=HWD/ABS(CS)                                                    HMTR0025
      DELR= (RO-RI)/2.0                                                 HMTR0026
      AVGR= (RO+RI)/2.0                                                 HMTR0027
      DO 20 IR=1,NZ                                                     HMTR0028
      R=F(DELR,AVGR,Z(IR))                                              HMTR0029
      CALL PMTRX(P,R)                                                   HMTR0030
      COEFF=DELR*WZ(IR)*R*WZ(ITHETA)*DELTHT                             HMTR0031
      DO 20 I=1,10                                                      HMTR0032
      DO 20 J=1,10                                                      HMTR0033
      DO 20 K=1,3                                                       HMTR0034
      DO 20 L=1,3                                                       HMTR0035
```

```
20    H(I,J)=H(I,J)+P(K,I)*S(K,L)*P(L,J)*COEFF                      HMTR0036
      L=0                                                           HMTR0037
      DO 30 J=1,10                                                  HMTR0038
      DO 30 I=1,J                                                   HMTR0039
      L=L+1                                                         HMTR0040
30    HV(L)=H(I,J)                                                  HMTR0041
      CALL SINV(HV,10,1.0E-03,INDCTR)                               HMTR0042
      L=0                                                           HMTR0043
      DO 40 J=1,10                                                  HMTR0044
      DO 40 I=1,J                                                   HMTR0045
      L=L+1                                                         HMTR0046
      H(I,J)=HV(L)                                                  HMTR0047
40    H(J,I)=H(I,J)                                                 HMTR0048
      RETURN                                                        HMTR0049
      END                                                           HMTR0050
```

```
      SUBROUTINE GMTRX(PI4,HWD,RI,G)                                    GMTR0000
C***********************************************************************GMTR0001
C SUBROUTINE GMTRX GENERATES G MATRIX                                   GMTR0002
C THIS SUBROUTINE IS USED IN FORMING THE HOLE SUBSTRUCTURE OF HOLFL     GMTR0003
C***********************************************************************GMTR0004
C COPYRIGHT (C) 1975 MASSACHUSETTS INSTITUTE OF TECHNOLOGY              GMTR0005
C AEROELASTIC AND STRUCTURES RESEARCH LABORATORY                        GMTR0006
C***********************************************************************GMTR0007
C FINITE ELEMENT APPLICATIONS TO USAF STRUCTURAL INTEGRITY PROBLEMS     GMTR0008
      DIMENSION Z(5),WZ(5)                                              GMTR0009
      DIMENSION G(10,12),P(3,10),RMINT(3,2),RL(2,12),THETAI(2)          GMTR0010
      COMMON/TRGFNT/CS,SS,CS2,SS2,CS4,SS4                               GMTR0011
      F(A,B,Z)=A*Z+B                                                    GMTR0012
      DATA WZ/.2369269         ,.4786287         ,.5688889              GMTR0013
     *.4786287         ,.2369269         /                              GMTR0014
      DATA Z/-.9061798         ,-.5384693         ,0.0E0,               GMTR0015
     *.5384693         ,.9061798         /,NZ/5/                        GMTR0016
      DO 10 J=1,12                                                      GMTR0017
      DO 10 I=1,10                                                      GMTR0018
   10 G(I,J)=0.0                                                        GMTR0019
      THETAI(1)=0.0                                                     GMTR0020
      THETAI(2)=PI4                                                     GMTR0021
C  BOUNDARY 1-2                                                         GMTR0022
      NODE1=1                                                           GMTR0023
      NODE2=NODE1+1                                                     GMTR0024
      DELTHT=PI4/2.0                                                    GMTR0025
      AVGTHT=DELTHT                                                     GMTR0026
      DO 20 ITHETA=1,NZ                                                 GMTR0027
      THETA=F(DELTHT,AVGTHT,Z(ITHETA))                                  GMTR0028
      CALL TRIG(THETA)                                                  GMTR0029
      RO=HWD/CS                                                         GMTR0030
      DS=RO/CS                                                          GMTR0031
      CALL PMTRX(P,RO)                                                  GMTR0032
      CALL MNMTRX(NODE1,NODE2,RMINT)                                    GMTR0033
      CALL LMTRX(NODE1,NODE2,THETAI,THETA,RI,RO,R,RL)                   GMTR0034
      DS=DS*WZ(ITHETA)*DELTHT                                           GMTR0035
```

```
      DO 20 I=1,10                                                      GMTR0036
      DO 20 J=1,12                                                      GMTR0037
      DO 20 K=1,3                                                       GMTR0038
      DO 20 L=1,2                                                       GMTR0039
   20 G(I,J)=G(I,J)+P(K,I)*RMTNT(K,L)*RL(L,J)*DS                        GMTR0040
C  BOUNDARY 2-3 & 6-1                                                   GMTR0041
      DO 30 IB=1,2                                                      GMTR0042
      THETA=THETAI(3-IB)                                                GMTR0043
      NODE1=2*(2-IB)+6*(IB-1)                                           GMTR0044
      NODE2=3*(2-IB)+(IB-1)                                             GMTR0045
      CALL TRIG(THETA)                                                  GMTR0046
      RO=HWD/ABS(CS)                                                    GMTR0047
      DELR=(RI-RO)/2.0*(-1.0)**(IB+1)                                   GMTR0048
      AVGR=(RO+RI)/2.0                                                  GMTR0049
      DS=ABS(DELR)                                                      GMTR0050
      DO 30 IR=1,N7                                                     GMTR0051
      R=F(DELR,AVGR,Z(IR))                                              GMTR0052
      CALL PMTRX(P,R)                                                   GMTR0053
      CALL MNMTRX(NODE1,NODE2,RMTNT)                                    GMTR0054
      CALL LMTRX(NODE1,NODE2,THETAI,THETA,RI,RO,R,RL)                   GMTR0055
      DDS=DS*WZ(IR)                                                     GMTR0056
      DO 30 I=1,10                                                      GMTR0057
      DO 30 J=1,12                                                      GMTR0058
      DO 30 K=1,3                                                       GMTR0059
      DO 30 L=1,2                                                       GMTR0060
   30 G(I,J)=G(I,J)+P(K,I)*RMTNT(K,L)*RL(L,J)*DDS                       GMTR0061
C  BOUNDARY I-(I+1)                                                     GMTR0062
      DTHT=PI4/3.0                                                      GMTR0063
      THETAI(1)=PI4                                                     GMTR0064
      NODE1=2                                                           GMTR0065
      DO 50 ITHETA=1,3                                                  GMTR0066
      NODE1=NODE1+1                                                     GMTR0067
      NODE2=NODE1+1                                                     GMTR0068
      THETAI(2)=THETAI(1)-DTHT                                          GMTR0069
      DELTHT=(THETAI(2)-THETAI(1))/2.0                                  GMTR0070
      AVGTHT=(THETAI(2)+THETAI(1))/2.0                                  GMTR0071
```

```
      DS=-DELTHT*.1                                                      GMTR0072
      DO 40 ITHT=1,NZ                                                    GMTR0073
      THETA=F(DELTHT,AVGTHT,Z(ITHT))                                     GMTR0074
      CALL TRIG(THETA)                                                   GMTR0075
      CALL PMTRX(P,RI)                                                   GMTR0076
      CALL MNMTRX(NODE1,NODE2,PMINT)                                     GMTR0077
      CALL LMTRX(NODE1,NODE2,THETAI,THETA,RI,RO,R,RL)                    GMTR0078
      DDS=DS*WZ(ITHT)                                                    GMTR0079
      DO 40 I=1,12                                                       GMTR0080
      DO 40 J=1,12                                                       GMTR0081
      DO 40 K=1,3                                                        GMTR0082
      DO 40 L=1,3                                                        GMTR0083
   40 G(I,J)=G(I,J)+P(K,I)*PMINT(K,L)*RL(L,J)*DDS                        GMTR0084
   50 THETAI(1)=THETAI(2)                                                GMTR0085
      RETURN                                                             GMTR0086
      END                                                                GMTR0087
```

```
      SUBROUTINE PMTRX(P,R)                                             PMTR0000
C***********************************************************************PMTR0001
C SUBROUTINE PMTRX GENERATES P MATRIX                                   PMTR0002
C THIS SUBROUTINE IS USED IN FORMING THE HOLE SUBSTRUCTURE OF HOLFL     PMTR0003
C***********************************************************************PMTR0004
C COPYRIGHT (C) 1975 MASSACHUSETTS INSTITUTE OF TECHNOLOGY              PMTR0005
C AEROELASTIC AND STRUCTURES RESEARCH LABORATORY                        PMTR0006
C***********************************************************************PMTR0007
C FINITE ELEMENT APPLICATIONS TO USAF STRUCTURAL INTEGRITY PROBLEMS     PMTR0008
      DIMENSION P(3,10),A(3,10)                                         PMTR0009
      COMMON/TRGFNT/CS,SS,CS2,SS2,CS4,SS4                               PMTR0010
      DATA A/-2.0,2.0,2.0,-6.0,6.0,-6.0,0.0,12.0,6.0,-4.0,0.0,-2.0,     PMTR0011
     W       -2.0,2.0,2.0,-6.0,6.0,-6.0,0.0,12.0,6.0,-4.0,0.0,-2.0,     PMTR0012
     *1.0,-1.0,0.0,2.0,2.0,0.0/                                         PMTR0013
      R2=R*R                                                            PMTR0014
      R4=R2*R2                                                          PMTR0015
      DO 10 I=1,3                                                       PMTR0016
      P(I,1)=1.0                                                        PMTR0017
   10 P(I,5)=1.0                                                        PMTR0018
      COEFF=1.0/R4                                                      PMTR0019
      DO 20 I=1,3                                                       PMTR0020
      P(I,2)=COEFF                                                      PMTR0021
   20 P(I,6)=COEFF                                                      PMTR0022
      DO 30 I=1,3                                                       PMTR0023
      P(I,3)=R2                                                         PMTR0024
   30 P(I,7)=R2                                                         PMTR0025
      COEFF=1.0/R2                                                      PMTR0026
      DO 40 I=1,3                                                       PMTR0027
      P(I,4)=COEFF                                                      PMTR0028
   40 P(I,8)=COEFF                                                      PMTR0029
C  LAST TWO BETAS:                                                      PMTR0030
      DO 50 I=1,3                                                       PMTR0031
      P(I,9)=COEFF                                                      PMTR0032
   50 P(I,10)=1.0                                                       PMTR0033
C  INTRODUCE TRIG FUNCTIONS                                             PMTR0034
      DO 60 J=1,6                                                       PMTR0035
```

```
      JP4=J+4                                                           PMTR0036
      P(3.J)=P(3.J)*SS2                                                 PMTR0037
      P(3.JP4)=-P(3.JP4)*CS2                                            PMTR0038
      DO 60 I=1.2                                                       PMTR0039
      P(I.J)=P(I.J)*CS2                                                 PMTR0040
   60 P(I.JP4)=P(I.JP4)*SS2                                             PMTR0041
C  INTRODUCE A MATRIX                                                   PMTR0042
      DO 70 I=1.3                                                       PMTR0043
      DO 70 J=1.10                                                      PMTR0044
   70 P(I.J)=P(I.J)*A(I.J)                                              PMTR0045
      RETURN                                                            PMTR0046
      END                                                               PMTR0047
```

```
      SUBROUTINE TRIG(THETA)
C***********************************************************************
C SUBROUTINE TRIG EVALUATES TRIGONOMETRIC FUNCTIONS
C THIS SUBROUTINE IS USED IN FORMING THE HOLE SUBSTRUCTURE OF HOLEL
C***********************************************************************
C COPYRIGHT (C) 1975 MASSACHUSETTS INSTITUTE OF TECHNOLOGY
C AEROELASTIC AND STRUCTURES RESEARCH LABORATORY
C***********************************************************************
C FINITE ELEMENT APPLICATIONS TO USAF STRUCTURAL INTEGRITY PROBLEMS
      COMMON/TRGENT/CS,SS,CS2,SS2,CS4,SS4
      CS= COS(THETA)
      SS= SIN(THETA)
      CS2=1.0E0-2.0E0*SS*SS
      SS2=2.0E0*SS*CS
      CS4=1.0E0-2.0E0*SS2*SS2
      SS4=2.0E0*CS2*SS2
      RETURN
      END
```

```
      SUBROUTINE LMTRX(NODE1,NODE2,THETAI,THETA,RI,RO,R,RL)             LMTR0000
C***********************************************************************LMTR0001
C SUBROUTINE LMTRX GENERATES L MATRIX                                   LMTR0002
C THIS SUBROUTINE IS USED IN FORMING THE HOLE SUBSTRUCTURE OF HOLFL     LMTR0003
C***********************************************************************LMTR0004
C COPYRIGHT (C) 1975 MASSACHUSETTS INSTITUTE OF TECHNOLOGY              LMTR0005
C AEROELASTIC AND STRUCTURES RESEARCH LABORATORY                        LMTR0006
C***********************************************************************LMTR0007
C FINITE ELEMENT APPLICATIONS TO USAF STRUCTURAL INTEGRITY PROBLEMS     LMTR0008
      DIMENSION RL(2,12),THETAI(2)                                      LMTR0009
      DO 10 I=1,2                                                       LMTR0010
      DO 10 J=1,12                                                      LMTR0011
 10   RL(I,J)=0.0                                                       LMTR0012
      DELR=RO-RI                                                        LMTR0013
      DELTHT=THETAI(1)-THETAI(2)                                        LMTR0014
      GO TO (1,2,3,4,5,6),NODE1                                         LMTR0015
 1    RL(1,1)=(THETA-THETAI(2))/DELTHT                                  LMTR0016
      RL(2,2)=RL(1,1)                                                   LMTR0017
      RL(1,3)=1.0-RL(1,1)                                               LMTR0018
      RL(2,4)=RL(1,3)                                                   LMTR0019
      GO TO 7                                                           LMTR0020
 2    RL(1,3)=(R-RI)/DELR                                               LMTR0021
      RL(2,4)=RL(1,3)                                                   LMTR0022
      RL(1,5)=1.0-RL(1,3)                                               LMTR0023
      RL(2,6)=RL(1,5)                                                   LMTR0024
      GO TO 7                                                           LMTR0025
 3    RL(1,5)=(THETA-THETAI(2))/DELTHT                                  LMTR0026
      RL(2,6)=RL(1,5)                                                   LMTR0027
      RL(1,7)=1.0-RL(1,5)                                               LMTR0028
      RL(2,8)=RL(1,7)                                                   LMTR0029
      GO TO 7                                                           LMTR0030
 4    RL(1,7)=(THETA-THETAI(2))/DELTHT                                  LMTR0031
      RL(2,8)=RL(1,7)                                                   LMTR0032
      RL(1,9)=1.0-RL(1,7)                                               LMTR0033
      RL(2,10)=RL(1,9)                                                  LMTR0034
      GO TO 7                                                           LMTR0035
```

```
5     RL(1,9)=(THETA-THETAI(2))/DELTHT                                LMTR0036
      RL(2,10)=RL(1,9)                                                LMTR0037
      RL(1,11)=1.0-RL(1,9)                                            LMTR0038
      RL(2,12)=RL(1,11)                                               LMTR0039
      GO TO 7                                                         LMTR0040
6     RL(1,11)=(RO-R)/DELR                                            LMTR0041
      RL(2,12)=RL(1,11)                                               LMTR0042
      RL(1,1)=1.0-RL(1,11)                                            LMTR0043
      RL(2,2)=RL(1,1)                                                 LMTR0044
7     CONTINUE                                                        LMTR0045
      RETURN                                                          LMTR0046
      END                                                             LMTR0047
```

```
      SUBROUTINE MNMTRX(NODE1,NODE2,RMTNT)                              MNMT0000
C***********************************************************************MNMT0001
C SUBROUTINE MNMTRX CREATES M AND N MATRICES AND FORMS THEIR PRODUCT    MNMT0002
C THIS SUBROUTINE IS USED IN FORMING THE HOLE SUBSTRUCTURE OF HOLEL     MNMT0003
C***********************************************************************MNMT0004
C COPYRIGHT (C) 1975 MASSACHUSETTS INSTITUTE OF TECHNOLOGY              MNMT0005
C AEROELASTIC AND STRUCTURES RESEARCH LABORATORY                        MNMT0006
C***********************************************************************MNMT0007
C FINITE ELEMENT APPLICATIONS TO USAF STRUCTURAL INTEGRITY PROBLEMS     MNMT0008
      DIMENSION RMTNT(3,2),RN(2,3),RM(3,3)                              MNMT0009
      COMMON/TRGFNT/CS,SS,CS2,SS2,CS4,SS4                               MNMT0010
      REAL NX,NY                                                        MNMT0011
      CSCS=CS*CS                                                        MNMT0012
      SSSS=SS*SS                                                        MNMT0013
      SSCS=SS*CS                                                        MNMT0014
      RM(1,1)=CSCS                                                      MNMT0015
      RM(2,1)=SSSS                                                      MNMT0016
      RM(3,1)=SSCS                                                      MNMT0017
      RM(1,2)=SSSS                                                      MNMT0018
      RM(2,2)=CSCS                                                      MNMT0019
      RM(3,2)=-SSCS                                                     MNMT0020
      RM(1,3)=-2.0*SSCS                                                 MNMT0021
      RM(2,3)=-RM(1,3)                                                  MNMT0022
      RM(3,3)=CSCS-SSSS                                                 MNMT0023
C  DETERMINE BOUNDARY SEGMENT                                           MNMT0024
      IF( NODE1 .EQ. 6 .AND. NODE2 .EQ. 1 ) GO TO 40                    MNMT0025
      IF( NODE1 .EQ. 1 .AND. NODE2 .EQ. 2 ) GO TO 10                    MNMT0026
      IF( NODE1 .EQ. 2 .AND. NODE2 .EQ. 3 ) GO TO 20                    MNMT0027
      IF( NODE1 .GE. 3 .AND. NODE2 .LE. 6 ) GO TO 30                    MNMT0028
 10   NX=1.0                                                            MNMT0029
      NY=0.0                                                            MNMT0030
      GO TO 50                                                          MNMT0031
 20   NX=-SS                                                            MNMT0032
      NY=CS                                                             MNMT0033
      GO TO 50                                                          MNMT0034
 30   NX=-CS                                                            MNMT0035
```

```
      NY=-SS                                                            MNMT0036
      GO TO 50                                                          MNMT0037
   40 NX=0.0                                                            MNMT0038
      NY=-1.0                                                           MNMT0039
   50 RN(1,1)=NX                                                        MNMT0040
      RN(1,2)=0.0                                                       MNMT0041
      RN(1,3)=NY                                                        MNMT0042
      RN(2,1)=0.0                                                       MNMT0043
      RN(2,2)=NY                                                        MNMT0044
      RN(2,3)=NX                                                        MNMT0045
      DO 60 I=1,3                                                       MNMT0046
      DO 60 J=1,2                                                       MNMT0047
      RMTNT(I,J)=0.0                                                    MNMT0048
      DO 60 K=1,3                                                       MNMT0049
   60 RMTNT(I,J)=RMTNT(I,J)+RM(K,I)*RN(J,K)                             MNMT0050
      RETURN                                                            MNMT0051
      END                                                               MNMT0052
```

```
C  TEST MAIN - CONTRACT 81739  PROBLEMS 1 & 2                            PAN0001
C*********************************************************************** PAN0002
C COPYRIGHT (C) 1975 MASSACHUSETTS INSTITUTE OF TECHNOLOGY               PAN0003
C AEROELASTIC AND STRUCTURES RESEARCH LABORATORY                         PAN0004
C*********************************************************************** PAN0005
C FINITE ELEMENT APPLICATIONS TO USAF STRUCTURAL INTEGRITY PROBLEMS      PAN0006
C.........INPUT DATA FOR PROBLEM #1&2.................................... PAN0007
C SET         VARIABLES IN SET                     FORMAT FOR EACH CARD   PAN0008
C  1          WIDTH,LENGTH,THK,STFFCT,PRESS,RI,IOFFST         (6E10.3,I5) PAN0009
C  2          A(1),A(2),IPOS(1),IPOS(2)                     (2E10.3,2I5) PAN0010
C  3          E,GNU                                             (2E10.3) PAN0011
C  4          KT1,KT2,KT3                                          (3I5) PAN0012
C =====DEFINITIONS OF INPUT VARIABLES==================================== PAN0013
C WIDTH = WIDTH OF PLATE;  LENGTH = LENGTH OF PLATE;                     PAN0014
C THK = THICKNESS OF PLATE;  STFFCT = STIFFNESS FACTOR;                  PAN0015
C PRESS = FARFIELD LOADING;  RI = RADIUS OF HOLE;                        PAN0016
C IOFFSET = OFFSET INDICATOR;  A(1) = LENGTH OF CRACK ONE;               PAN0017
C A(2) = LENGTH OF CRACK TWO; IPOS(1) = INITIAL POSITION OF CRACK ONE;   PAN0018
C IPOS(2) = FINAL POSITION OF CRACK ONE;  E = YOUNGS MODULUS;            PAN0019
C GNU = POISSONS RATIO;KT1 = PRINT CONTROL FOR FEABL OPTIONAL PRINT;     PAN0020
C KT2 = PRINT CONTROL FOR OPTIONAL FARFLD PRINTING;                      PAN0021
C KT3 = PRINT CONTROL FOR OPTIONAL RING PRINTING;                        PAN0022
C*********************************************************************** PAN0023
      DIMENSION RFF(10000),IFF(10000),REAL(8000),INTGR(8000),RRNG(12000) PAN0024
     #,IRNG(12000),S(3,3),C(3,3),IBEGIN(6),IEND(6),ISZ(2),ISZSS(3),      PAN0025
     #ANGLE(2,2),A(2),IPOS(2),ISZFF(3),IBGFF(6),ISAVE(6),ELQ(18),RK(2)   PAN0026
      REAL LENGTH                                                        PAN0027
      INTEGER CNODE,PRNTSV                                               PAN0028
      COMMON/IO/KR,KW,KP,KT1,KT2,KT3                                     PAN0029
      COMMON/SIZE/NET,NDT                                                PAN0030
      COMMON/BEGIN/ICON,IKOUNT,ILNZ,IMASTR,IQ,IK                         PAN0031
      COMMON/END/LCON,LKOUNT,LLNZ,LMASTR,LQ,LK                           PAN0032
      COMMON/SIZESS/NETSS,NDTSS,NIDSS                                    PAN0033
      COMMON/BEGSS/IBGSS(6)                                              PAN0034
      COMMON/MTLPRM/SMU,ETA                                              PAN0035
      COMMON/KI/BCRK(2,2,18)                                             PAN0036
```

```
      EQUIVALENCE (RFF(1),IFF(1)),(REAL(1),INTGR(1)),(IBEGIN(1),ICON),    PAN0037
     *(IEND(1),LCON),(ISZ(1),NET),(ISZSS(1),NETSS),(RRNG(1),IRNG(1))      PAN0038
      DATA NSZFF/10000/,NSZRNG/12000/,NSZMN/8000/                         PAN0039
      DATA PI/3.141593/,S/1.0,3*0.0,1.0,4*0.0/,C/1.0,3*0.0,1.0,4*0.0/     PAN0040
      KR=5                                                                PAN0041
      KW=6                                                                PAN0042
      DTHETA=PI/12.0                                                      PAN0043
      SQRTPI=SQRT(PI)                                                     PAN0044
C   INPUT DATA:                                                           PAN0045
      READ(KR,5000) WIDTH,LENGTH,THK,STFFCT,PRESS,RI,IOFFST               PAN0046
 5000 FORMAT(6E10.3,I5)                                                   PAN0047
      READ(KR,5001) A(1),A(2),IPOS(1),IPOS(2)                             PAN0048
 5001 FORMAT(2E10.3,2I5)                                                  PAN0049
      READ(KR,5002) E,GNU                                                 PAN0050
 5002 FORMAT(2E10.3)                                                      PAN0051
      READ(KR,5003) KT1,KT2,KT3                                           PAN0052
 5003 FORMAT(3I5)                                                         PAN0053
      ANGLE(1,1)=(IPOS(1)-1)*15.0                                         PAN0054
      ANGLE(1,2)=(IPOS(2)-1)*15.0                                         PAN0055
      ANGLE(2,1)=ANGLE(1,1)+180.0                                         PAN0056
      ANGLE(2,2)=ANGLE(1,2)+180.0                                         PAN0057
      WRITE(KW,6000) WIDTH,LENGTH,THK,STFFCT,PRESS,RI,IOFFST              PAN0058
 6000 FORMAT(1H1,58X,14HCONTRACT 81739,//57X,18HPROBLEMS NO. 1 & 2,///,6 PAN0059
     *0X,11H INPUT DATA,///,16H PLATE GEOMETRY:,//,13H PLATE WIDTH=,E12. PAN0060
     *5,/,14H PLATE LENGTH=,E12.5,/,17H PLATE THICKNESS=,E12.5,/,24H PLA PAN0061
     *TE STIFFENER FACTOR=,F8.3,/,25H FAR FIELD LOADING (PSI)=,E12.5,/,1 PAN0062
     *6H RADIUS OF HOLE=,E12.5,/,23H HOLE OFFSET INDICATOR=,I3,27H(=0, H PAN0063
     *OLE REMAINS CENTERED),//,12H CRACK DATA:,/)                         PAN0064
      NCRK=0                                                              PAN0065
      DO 10 I=1,2                                                         PAN0066
   10 IF( A(I) .GT. 0.0) NCRK=NCRK+1                                      PAN0067
      DO 15 I=1,NCRK                                                      PAN0068
   15 WRITE(KW,6001) I,A(I),(IPOS(J),ANGLE(I,J),J=1,2)                    PAN0069
 6001 FORMAT(11H CRACK NO. ,I3,17H :  CRACK LENGTH=,E12.5,/,18X,23HINITI PAN0070
     *AL CRACK POSITION=,I3,1H(,F8.3,9H DEGREES),/,20X,21HFINAL CRACK PO PAN0071
     *SITION=,I3,1H(,F8.3,9H DEGREES),/)                                  PAN0072
```

```
      WRITE(KW,6002) E,GNU
 6002 FORMAT(21HOMATERIAL PROPERTIES:,//,20H YOUNGS MODULUS (E)=,E12.5,/
     W,16H POISSONS RATIO=,E12.5,///)
C  CALCULATE S&C MATRICES:
      C(2,1)=GNU
      C(3,3)=(1.0-GNU)/2.0
      S(2,1)=-GNU
      S(3,3)=(1.0+GNU)*2.0
      SMU=E/(2.0*(1.0+GNU))
      ETA=(3.0-GNU)/(1.0+GNU)
      GNU=E/(1.0-GNU*GNU)
      DO 20 I=1,3
      DO 20 J=1,I
      S(I,J)=S(I,J)/E
      S(J,I)=S(I,J)
      C(I,J)=C(I,J)*GNU
   20 C(J,I)=C(I,J)
C  DETERMINE NUMBER OF QUADRILATERALS NEEDED ACROSS WIDTH:
      NELW=WIDTH/(4.0*RI)
C  INSURE THAT NELW IS EVEN VALUED:
      I=NELW/2
      I=I*2
      IF(NELW .NE. I ) NELW=NELW-1
      W=WIDTH/NELW
      RATIO=W/RI
C  DETERMINE STIFFENER THICKNESS:
      STFTHK=THK+STFFCT*WIDTH*THK/W
C  CENTER NODE REFERENCE (INPUT FOR SUBROUTINE CZONE):
      CNODE=NELW/2+1
      LIMIT=2
      IF(STFFCT .NE. 0.) LIMIT=3
C  HOLE WILL INITIALLY BE PLACED IN CENTER OF PLATE.   IF IOFFST IS NOT
C  EQUAL TO 0 THE HOLE CENTER WILL BE MOVED TO THE RIGHT IN INCREMENTS O
C  W AS FAR AS IS POSSIBLE
C
C  PRINT OUT PROGRAM STATISTICS TO DATE:
```

```
      IF(KT1 .EQ. KW) GO TO 30                                           PAN0109
      WRITE(KW,6003) NELW,W,RATIO,CNODE,STFTHK,SMU,ETA                   PAN0110
 6003 FORMAT(20H0PROGRAM STATISTICS:,//,33H NUMBER OF ELEMENTS ACROSS WI PAN0111
     *DTH=,I4,/,27H QUADRILATERAL DIMENSION W=,E12.5,/,30H HOLE ELEMENT  PAN0112
     *DIMENSION RATIO=,F8.3,/,23H CENTER NODE REFERENCE=,I4,/,21H STIFFE PAN0113
     *NER THICKNESS=,E12.5,//,21H MATERIAL PARAMETERS:,/,5H SMU=,E12.5,/ PAN0114
     *,5H ETA=,E12.5,//,10H S MATRIX:,/)                                 PAN0115
      WRITE(KW,6004) ((S(I,J),J=1,3),I=1,3)                              PAN0116
 6004 FORMAT(1H ,3E12.5)                                                 PAN0117
      WRITE(KW,6005)                                                     PAN0118
 6005 FORMAT(10H0C MATRIX:)                                              PAN0119
      WRITE(KW,6004) ((C(I,J),J=1,3),I=1,3)                              PAN0120
 30   CONTINUE                                                           PAN0121
      E=0.0                                                              PAN0122
C  OBTAIN FAR FIELD SUBSTRUCTURE (CHESIRE CAT):                          PAN0123
 32   CONTINUE                                                           PAN0124
      PRNTSV=KT1                                                         PAN0125
      KT1=KT2                                                            PAN0126
      RO=(1.0+DTHETA)**4*RI                                              PAN0127
      INDCTR=0                                                           PAN0128
      CALL FARFLD(WIDTH,LENGTH,NELW,THK,STFTHK,C,S,CNODE,RO,PRESS,INDCTR PAN0129
     *,NSZFF,RFF,IFF,NSZMN,REAL,INTGR)                                   PAN0130
      JLOC=IPOS(1)                                                       PAN0131
      KT1=PRNTSV                                                         PAN0132
      DO 35 I=1,3                                                        PAN0133
 35   ISZFF(I)=ISZSS(I)                                                  PAN0134
      DO 36 I=1,6                                                        PAN0135
 36   IBGFF(I)=IBGSS(I)                                                  PAN0136
C  SET UP CRACK PROBLEM:                                                 PAN0137
      NET=2                                                              PAN0138
      ND7=96+2*NCRK                                                      PAN0139
      IPNTR=2*NCRK+146                                                   PAN0140
      CALL SETUP(NSZMN,1,IPNTR,REAL,INTGR)                               PAN0141
C  SET UP ASSEMBLY LIST:                                                 PAN0142
C  ELEMENT NO. 1 IS THE FAR FIELD SUBSTRUCTURE (CHESIRE CAT)             PAN0143
      IPNTR=IMASTR+2                                                     PAN0144
```

```
      INTGR(IMASTR)=IPNTR                                              PAN0145
      DO 40 I=1,48                                                     PAN0146
      INTGR(IPNTR)=I                                                   PAN0147
   40 IPNTR=IPNTR+1                                                    PAN0148
C  ELEMENT NO. 2 IS THE CRACKED RING                                   PAN0149
      INTGR(IMASTR+1)=IPNTR                                            PAN0150
   45 IPNTR=INTGR(IMASTR+1)                                            PAN0151
      DO 50 I=49,NDT                                                   PAN0152
      INTGR(IPNTR)=I                                                   PAN0153
   50 IPNTR=IPNTR+1                                                    PAN0154
      J=(4+JLOC)*2-1                                                   PAN0155
      IF(JLOC .GE. 21) J=(JLOC-20)*2-1                                 PAN0156
      DO 60 I=J,48                                                     PAN0157
      INTGR(IPNTR)=I                                                   PAN0158
   60 IPNTR=IPNTR+1                                                    PAN0159
      J=J-1                                                            PAN0160
      IF(JLOC .EQ. 21) GO TO 75                                        PAN0161
      DO 70 I=1,J                                                      PAN0162
      INTGR(IPNTR)=I                                                   PAN0163
   70 IPNTR=IPNTR+1                                                    PAN0164
   75 CONTINUE                                                         PAN0165
      IF(KTI .EQ. KW) GO TO 80                                         PAN0166
      J=IMASTR+1                                                       PAN0167
      WRITE(KW,6006) (INTGR(I),I=IMASTR,J)                             PAN0168
 6006 FORMAT(22H0MASTER ASSEMBLY LIST:,/,59H POINTER FOR FAR FIELD SUBST PAN0169
     *RUCTURED ELEMENT (CHESIRE CAT)=,I5,/,26H POINTER FOR CRACKED RING= PAN0170
     *,I5,/)                                                           PAN0171
      J=J+1                                                            PAN0172
      IPNTR=J+47                                                       PAN0173
      WRITE(KW,6007) (INTGR(I),I=J,IPNTR)                              PAN0174
 6007 FORMAT(44H0FAR FIELD SUBSTRUCTURED ELEMENT D.O.F.S:    ,16I5,/44X,1 PAN0175
     *6I5,/,44X,16I5)                                                  PAN0176
      IPNTR=IPNTR+1                                                    PAN0177
      J=IPNTR+95+NCRK*2                                                PAN0178
      WRITE(KW,6008) (INTGR(I),I=IPNTR,J)                              PAN0179
 6008 FORMAT(25H0CRACKED RING D.O.F.S:    ,16I5,/,25X,16I5,/,25X,16I5,/,2 PAN0180
```

```
     *5X,16I5,/,25X,16I5,/,25X,16I5,/,25X,16I5)                          PAN0181
  80   CALL OWK(NSZMN,REAL,INTGR)                                         PAN0182
C  ASSEMBLE FAR FIELD SUBSTRUCTURE:                                       PAN0183
       DO 85 I=1,3                                                        PAN0184
  85   ISZSS(I)=ISZFF(I)                                                  PAN0185
       DO 86 I=1,6                                                        PAN0186
  86   IBGSS(I)=IBGFF(I)                                                  PAN0187
       CALL ASMSUB(1,RFF,IFF,REAL,INTGR)                                  PAN0188
C  COMPUTE AND ASSEMBLE CRACKED RING ELEMENT:                             PAN0189
       THTCRK=(JLOC-1)*DTHETA                                             PAN0190
       INDCTR=0                                                           PAN0191
       PHNTSV=KT1                                                         PAN0192
       KT1=KT3                                                            PAN0193
       CALL RING(RI,THK,THTCRK,NCRK,A(1),A(2),INDCTR,S,C,NSZRNG,RRNG,     PAN0194
     *IRNG)                                                               PAN0195
       KT1=PHNTSV                                                         PAN0196
       CALL ASMSUB(2,RRNG,IRNG,REAL,INTGR)                                PAN0197
C OBTAIN SOLUTION:                                                        PAN0198
       I=1                                                                PAN0199
       CALL FACT(I,REAL,INTGR)                                            PAN0200
       CALL SIMULQ(GNU,REAL,INTGR)                                        PAN0201
C  OBTAIN CRACKED RING INTERIOR DISPLACEMENTS:                            PAN0202
       CALL QBACK(2,RRNG,IRNG,REAL,INTGR)                                 PAN0203
       IF(NCRK .EQ. 0 ) GO TO 120                                         PAN0204
C  OBTAIN KI AND KII:                                                     PAN0205
       WRITE(KW,6012)                                                     PAN0206
 6012 FORMAT(1H0)                                                         PAN0207
       DO 90 I=1,6                                                        PAN0208
       ISAVE(I)=IBEGIN(I)                                                 PAN0209
  90   IBEGIN(I)=IBGSS(I)                                                 PAN0210
       THTCRK=THTCRK*180.0/PI                                             PAN0211
       DO 110 I=1,NCRK                                                    PAN0212
       CALL ATRACT(I*18,18,ELQ,RRNG,IRNG)                                 PAN0213
       DO 100 J=1,2                                                       PAN0214
       RK(J)=0.0                                                          PAN0215
       DO 100  K=1,18                                                     PAN0216
```

```
 100   RK(J)=RK(J)+BCKK(I,J,K)*ELQ(K)                                     PAN0217
       DO 105 J=1,2                                                       PAN0218
 105   RK(J)=RK(J)*SQRTPI                                                 PAN0219
       RK(2)=ABS(RK(2))                                                   PAN0220
       WRITE(KW,6010) I,THTCRK,(RK(J),J=1,2)                              PAN0221
 6010 FORMAT(10H CRACK NO.,I2,8H  ANGLE=,F8.3,8H     KI=,E12.5,8H    KII PAN0222
     *=,E12.5)                                                            PAN0223
 110   THTCRK=THTCRK+180.0                                                PAN0224
C   INCREMENT CRACK POSITION:                                             PAN0225
 120   JLOC=JLOC+1                                                        PAN0226
       DO 125 I=1,6                                                       PAN0227
 125   IBEGIN(I)=ISAVE(I)                                                 PAN0228
       IF(JLOC .LE. IPOS(2)) GO TO 45                                     PAN0229
C   INCREMENT HOLE POSITION:                                              PAN0230
       IF(IOFFST .EQ. 0) GO TO 130                                        PAN0231
       IF(CNODE .LE. LIMIT) GO TO 130                                     PAN0232
       E=E+W                                                              PAN0233
       CNODE=CNODE+1                                                      PAN0234
       WRITE(KW,6011) E                                                   PAN0235
 6011 FORMAT(1H1,53X,24H INCREMENT HOLE POSITION,//,54X,13HHOLE OFFSET = PAN0236
     *E12.5,///)                                                          PAN0237
       GO TO 32                                                           PAN0238
 130   CONTINUE                                                           PAN0239
       STOP                                                               PAN0240
       END                                                                PAN0241
```

```
      SUBROUTINE LUG(WIDTH,LENGTH,NELW,THK,STFTHK,C,NSIZE,RSS,ISS)      LUG 0000
C***********************************************************************LUG 0001
C SUBROUTINE LUG GENERATES A RECTANGULAR REGION WHICH IS USED BY        LUG 0002
C SUBROUTINES FARFLD, SBL AND DBL                                       LUG 0003
C***********************************************************************LUG 0004
C COPYRIGHT (C) 1975 MASSACHUSETTS INSTITUTE OF TECHNOLOGY              LUG 0005
C AEROELASTIC AND STRUCTURES RESEARCH LABORATORY                        LUG 0006
C***********************************************************************LUG 0007
C FINITE ELEMENT APPLICATIONS TO USAF STRUCTURAL INTEGRITY PROBLEMS     LUG 0008
      DIMENSION RSS(1),ISS(1),C(3,3),ELK(36),ELKSTF(36)                 LUG 0009
     *,ANGLE(4),TEMP(4),COORD(12),ND(4),ELQ(8)                          LUG 0010
      INTEGER ELND                                                      LUG 0011
      REAL LENGTH,L                                                     LUG 0012
      COMMON/IO/KR,KW,KP,KT1,KT2,KT3                                    LUG 0013
      COMMON/SIZE/NET,NDT                                               LUG 0014
      COMMON/BEGIN/IBEGIN(6)                                            LUG 0015
      COMMON/END/IEND(6)                                                LUG 0016
      COMMON/SIZESS/NETSS,NDTSS,NIDSS                                   LUG 0017
      COMMON/REGSS/IRGSS(6)                                             LUG 0018
      COMMON/STRSS/ B(3,9),BSTF(3,9)                                    LUG 0019
      DATA TEMP/4*0.0/                                                  LUG 0020
      DO 5 I=1,12                                                       LUG 0021
    5 COORD(I)=0.0                                                      LUG 0022
      W=WIDTH/NELW                                                      LUG 0023
      AR=5.0                                                            LUG 0024
      NELL=LENGTH/(W*AR)+1                                              LUG 0025
      NDW=NELW+1                                                        LUG 0026
      NDL=NELL+1                                                        LUG 0027
      NET=NELL*NELW                                                     LUG 0028
      NDT=(NDL+1)*NDW*2                                                 LUG 0029
      NIDSS=NDT-4*NDW                                                   LUG 0030
      IMASTR=NET*9                                                      LUG 0031
      L=LENGTH/NELL                                                     LUG 0032
      AR=L/W                                                            LUG 0033
      COORD(1)=W                                                        LUG 0034
      COORD(2)=L                                                        LUG 0035
```

```
      COORD(5)=L                                                        LUG 0036
      COORD(10)=W                                                       LUG 0037
      LMASTR=NOW*2                                                      LUG 0038
      IF( KT1 .EQ. KW ) GO TO 10                                        LUG 0039
      WRITE(KW,6000)WIDTH,LENGTH,THK,STFTHK,NELW                        LUG 0040
 6000 FORMAT(1H1,60X,9HENTRY LUG,//,23H SUBROUTINE INPUT DATA:,//,11H LULUG 0041
     *G WIDTH=,E12.5,/,12H LUG LENGTH=,E12.5,/,15H LUG THICKNESS=,E12.5,LUG 0042
     */,25H LUG STIFFENER THICKNESS=,E12.5,30H   ( LEFT SIDE STIFFENED OLUG 0043
     *NLY),/,33H NUMBER OF ELEMENTS ACROSS WIDTH=,I3,/)                 LUG 043A
      WRITE(KW,6001)NELL,NOW,NOL,NET,NOT,NIDSS,(COORD(I),I=1,12),AR     LUG 0044
 6001 FORMAT(27H0SUBROUTINE LUG STATISTICS:,//,37H NUMBER OF ELEMENTS ALLUG 0045
     *ONG LUG LENGTH=,I3,/,29H NUMBER OF NODES ALONG WIDTH=,I3,/,30H NUMLUG 0046
     *BER OF NODES ALONG LENGTH=,I3,/,23H TOTAL NUMBER ELEMENTS=,I3,/,25LUG 0047
     *H TOTAL NUMBER OF D.O.F.S=,/,63H NUMBER OF D.O.F.S TO BE CONDENSEDLUG 0048
     * OUT GLOBAL STIFFNESS MATRIX=,I3,/,21H ELEMENT COORDINATES:,12F8.3LUG 0049
     *,/,22H ELEMENT ASPECT RATIO=,F8.3)                                LUG 0050
   10 CALL SETUP(NSIZE,LMASTR,IMASTR,RSS,ISS)                           LUG 0051
      ELNO=0                                                            LUG 0052
      IMASTR=IBEGIN(4)                                                  LUG 0053
      DO 30 I=1,NELL                                                    LUG 0054
      INODE1=(I-1)*NOW+1                                                LUG 0055
      LNODE1=INODE1+NELW-1                                              LUG 0056
      DO 20 II=INODE1,LNODE1                                            LUG 0057
      IPNTR=IMASTR+NET+ELNO*8                                           LUG 0058
      ISS(IMASTR+ELNO)=IPNTR                                            LUG 0059
      ELNO=ELNO+1                                                       LUG 0060
      ISS(IPNTR+1)=II*2                                                 LUG 0061
      ISS(IPNTR)=ISS(IPNTR+1)-1                                         LUG 0062
      ISS(IPNTR+2)=ISS(IPNTR+1)+1                                       LUG 0063
   20 ISS(IPNTR+3)=ISS(IPNTR+2)+1                                       LUG 0064
      LNODE1=INODE1+NOW+NELW                                            LUG 0065
      DO 30 II=1,NELW                                                   LUG 0066
      ISS(IPNTR+5)=LNODE1*2                                             LUG 0067
      ISS(IPNTR+4)=ISS(IPNTR+5)-1                                       LUG 0068
      ISS(IPNTR+6)=ISS(IPNTR+4)-2                                       LUG 0069
      ISS(IPNTR+7)=ISS(IPNTR+6)+1                                       LUG 0070
      IPNTR=IPNTR-8                                                     LUG 0071
```

```
30    LNODE1=LNODE1-1                                                    LUG 0072
      IMASTR=IBEGIN(4)     +NET                                          LUG 0073
      LMASTR=IMASTR+NELW*8                                               LUG 0074
      INODE1=NDT-2*NDW+1                                                 LUG 0075
      LNODE1=1                                                           LUG 0076
      DO 35 I=INODE1,NDT                                                 LUG 0077
      DO 34 II=IMASTR,LMASTR                                             LUG 0078
      IF( ISS(II) .EQ. LNODE1 ) ISS(II)=I                                LUG 0079
34    CONTINUE                                                           LUG 0080
35    LNODE1=LNODE1+1                                                    LUG 0081
36    IF(KT1 .EQ. KW ) GO TO 60                                          LUG 0082
      WRITE(KW,6002)                                                     LUG 0083
6002  FORMAT(22H0MASTER ASSEMBLY LIST:,//,18H ELEMENT POINTERS:,/)       LUG 0084
      IMASTR=IBEGIN(4)                                                   LUG 0085
      LMASTR=IMASTR+NET-1                                                LUG 0086
      ELNO=0                                                             LUG 0087
      DO 40 I=IMASTR,LMASTR                                              LUG 0088
      ELNO=ELNO+1                                                        LUG 0089
40    WRITE(KW,6003)ELNO,ISS(I)                                          LUG 0090
6003  FORMAT(13H ELEMENT NO.=,I5,13H      POINTER=,I5)                   LUG 0091
      IMASTR=LMASTR+1                                                    LUG 0092
      WRITE(KW,6004)                                                     LUG 0093
6004  FORMAT(17H0ELEMENT D.O.F.S:,/)                                     LUG 0094
      DO 50 I=1,NET                                                      LUG 0095
      LMASTR=IMASTR+7                                                    LUG 0096
      WRITE(KW,6005)I,(ISS(J),J=IMASTR,LMASTR)                           LUG 0097
6005  FORMAT(12H ELEMENT NO.,I3,13H      D.O.F.S:,8I5)                   LUG 0098
50    IMASTR= IMASTR+8                                                   LUG 0099
60    CALL ORK(NSIZE,RSS,ISS)                                            LUG 0100
      CALL QUAD4(COORD,THK,TEMP,C,0,ND,ANGLE,ELQ,ELK,B,1,KW)             LUG 0101
      AR=STFTHK/THK                                                      LUG 0102
      DO 70 I=1,36                                                       LUG 0103
70    ELKSTF(I)=ELK(I)*AR                                                LUG 0104
      DO 80 I=1,3                                                        LUG 0105
      DO 80 J=1,9                                                        LUG 0106
80    BSTF(I,J)=B(I,J)*AR                                                LUG 0107
```

```
      NELWM1=NELW-1                                                     LUG 0108
      ELNO=0                                                            LUG 0109
      DO 100 IROW=1,NELL                                                LUG 0110
      ELNO=ELNO+1                                                       LUG 0111
      CALL ASMLTV(ELNO,8,ELKSTF,ELQ,RSS,ISS)                            LUG 0112
      IF(NELWM1 .LT. 2) GO TO 92                                        LUG 0113
      DO 90 IHEBE=2,NELWM1                                              LUG 0114
      ELNO=ELNO+1                                                       LUG 0115
90    CALL ASMLTV(ELNO,8,ELK,ELQ,RSS,ISS)                               LUG 0116
   92 ELNO=ELNO+1                                                       LUG 0117
  100 CALL ASMLTV(ELNO,8,ELKSTF,ELQ,RSS,ISS)                            LUG 0118
      LMASTR=NDW*2                                                      LUG 0119
      IMASTR=IBEGIN(5)-1                                                LUG 0120
      DO 110 I=1,LMASTR                                                 LUG 0121
      ISS(I)=I                                                          LUG 0122
110   RSS(IMASTR+I)=0.0                                                 LUG 0123
      CALL BCON(RSS,ISS)                                                LUG 0124
115   I=1                                                               LUG 0125
      CALL STACON(I,NIDSS,RSS,ISS)                                      LUG 0126
      DO 120 I=1,6                                                      LUG 0127
120   IBGSS(I)=IBEGIN(I)                                                LUG 0128
      NETSS=NET                                                         LUG 0129
      NDTSS=NDT                                                         LUG 0130
      IF( KT1 .EQ. KW) GO TO 130                                        LUG 0131
      WRITE(KW,6006)                                                    LUG 0132
6006  FORMAT(1H0,60X,8HEXIT LUG,//)                                     LUG 0133
130   CONTINUE                                                          LUG 0134
      RETURN                                                            LUG 0135
      END                                                               LUG 0136
```

```
      SUBROUTINE CZONE(WIDTH,NELW,THK,STFTHK,C,S,CNODE,RI,NSIZE,RCZONE, CZN 0000
     @ICZONE,RHOLE,IHOLE)                                               CZN 0001
C*********************************************************************CZN 0002
C SUBROUTINE CZONE GENERATES THE CENTRAL STRIP THAT CONTAINS THE HOLE   CZN 0003
C FOR USE IN THE FARFIELD MESH GENERATED BY SUBROUTINE FARFLD          CZN 0004
C*********************************************************************CZN 0005
C COPYRIGHT (C) 1975 MASSACHUSETTS INSTITUTE OF TECHNOLOGY             CZN 0006
C AEROELASTIC AND STRUCTURES RESEARCH LABORATORY                       CZN 0007
C*********************************************************************CZN 0008
C FINITE ELEMENT APPLICATIONS TO USAF STRUCTURAL INTEGRITY PROBLEMS     CZN 0009
      DIMENSION RCZONE(1),ICZONE(1),RHOLE(1),IHOLE(1),C(3,3),S(3,3),ELK(CZN 0010
     @36),ELKSTF(36),ANGLE(4),TEMP(4),ND(4),COORD(12),ELO(8),NOTASM(4), CZN 0011
     @IELSTF(4),IDOF(8),RHOLE(6,3,13)                                   CZN 0012
      INTEGER CNODE,ELNO                                                CZN 0013
      COMMON/IO/KR,KW,KP,KT1,KT2,KT3                                    CZN 0014
      COMMON/SIZE/NET,NDT                                               CZN 0015
      COMMON/BEGIN/IBEGIN(6)                                            CZN 0016
      COMMON/END/IEND(6)                                                CZN 0017
      COMMON/SIZESS/ISZ(3)                                              CZN 0018
      COMMON/BEGSS/IBGSS(6)                                             CZN 0019
      COMMON/STRSS/B(3,9),BSTF(3,9)                                     CZN 0020
      DATA TEMP/4*0.0/                                                  CZN 0021
      DO 5 I=1,12                                                       CZN 0022
    5 COORD(I)=0.0                                                      CZN 0023
      W=WIDTH/NELW                                                      CZN 0024
      NDW=NELW+1                                                        CZN 0025
      IF( CNODE .GT. 1 .AND. CNODE .LT. NDW ) GO TO 4                   CZN 0026
      WRITE(KW,6008) CNODE,NDW                                          CZN 0027
 6008 FORMAT(60H0***** PROGRAM INTERUPT INITIATED BY SUBROUTINE CZONE **CZN 0028
     @****,///,34H CENTER NODE REFERENCE PARAMETER =,I4,35H IS OUT OF RACZN 0029
     @NGE OF E WIDTH (NELW).,/,44H CNODE MUST BE GREATER THAN 1 AND LESSCZN 0030
     @ THAN ,I4,//,53H ***** EXECUTION TERMINATED IN SUBROUTINE CZONE **CZN 0031
     @***)                                                              CZN 031A
      STOP                                                              CZN 0032
    4 NET=1+2*NELW                                                      CZN 0033
      NDT=2*(3*NDW+24)                                                  CZN 0034
      NIDSS=NDT-4*NDW-48                                                CZN 0035
```

```
      NOTASM(1)=CNODE-1                                                  CZN 0036
      NOTASM(2)=CNODE                                                    CZN 0037
      NOTASM(3)=CNODE-1+NELW                                             CZN 0038
      NOTASM(4)=CNODE+NELW                                               CZN 0039
      IELSTF(1)=1                                                        CZN 0040
      IELSTF(2)=NELW+1                                                   CZN 0041
      IELSTF(3)=NELW                                                     CZN 0042
      IELSTF(4)=2*NELW                                                   CZN 0043
      COORD(1)=W                                                         CZN 0044
      COORD(2)=W                                                         CZN 0045
      COORD(5)=W                                                         CZN 0046
      COORD(10)=W                                                        CZN 0047
      LMASTR=(NET-1)*9+65                                                CZN 0048
      IF(KTI .EQ. KW) GO TO 10                                           CZN 0049
      WRITE(KW,6000) WIDTH,NELW,THK,STFTHK,CNODE,RI                      CZN 0050
 6000 FORMAT(1H1,59X,11HENTRY CZONE,///,23H SUBROUTINE INPUT DATA:,//,13CZN 0051
     @H PLATE WIDTH=,E12.5,/,33H NUMBER OF ELEMENTS ACROSS WIDTH=,I4,/,1CZN 0052
     @7H PLATE THICKNESS=,E12.5,/,21H STIFFENER THICKNESS=,F12.5,/,28H HCZN 0053
     @OLE CENTER NODE REFERENCE=,I3,/,27H HOLE ELEMENT INPUT RADIUS=,E12CZN 0054
     @.5)                                                                CZN 054A
      WRITE(KW,6001) NDW,NIDSS,(NOTASM(I),I=1,4),(IELSTF(I),I=1,4),(COORCZN 0055
     @D(I),I=1,12)                                                       CZN 0056
 6001 FORMAT(29H0SUBROUTINE CZONE STATISTICS:,//,30H NUMBER OF NODES ACRCZN 0057
     @OSS WIDTH=,I4,/,75H NUMBER OF INTERNAL D.O.F.S TO BE CONDENSED OUTCZN 0058
     @ OF GLOBAL STIFFNESS MATRIX=,I4,/,30H ELEMENTS NOT TO BE ASSEMBLEDCZN 0059
     @:,4I5,/,26H ELEMENTS TO BE STIFFENED:,4I5,/,21H ELEMENT COORDINATECZN 0060
     @S:,12F8.3)                                                         CZN 060A
 10   CALL SETUP(NSIZE,2,LMASTR,RCZONE,ICZONE)                           CZN 0061
      ELNO=0                                                             CZN 0062
      IMASTR=IBEGIN(4)                                                   CZN 0063
      INODE=NDW+1                                                        CZN 0064
      DO 30 IROW=1,2                                                     CZN 0065
      IF(IROW .EQ. 2) INODE=1                                            CZN 0066
      LNODE=INODE+NELW-1                                                 CZN 0067
      DO 20 I=INODE,LNODE                                                CZN 0068
      IPNTR=IMASTR+NET+ELNO*8                                            CZN 0069
      ICZONE(IMASTR+ELNO)=IPNTR                                          CZN 0070
      ELNO=ELNO+1                                                        CZN 0071
```

```
      ICZONE(IPNTR+1)=I*2                                                CZN 0072
      ICZONE(IPNTR)=ICZONE(IPNTR+1)-1                                    CZN 0073
      ICZONE(IPNTR+2)=ICZONE(IPNTR+1)+1                                  CZN 0074
   20 ICZONE(IPNTR+3)=ICZONE(IPNTR+2)+1                                  CZN 0075
      LNODE=NDW                                                          CZN 0076
      IF(IROW .EQ. 2) LNODE=3*NDW                                        CZN 0077
      DO 30 I=1,NELW                                                     CZN 0078
      ICZONE(IPNTR+5)=LNODE*2                                            CZN 0079
      ICZONE(IPNTR+4)=ICZONE(IPNTR+5)-1                                  CZN 0080
      ICZONE(IPNTR+6)=ICZONE(IPNTR+4)-2                                  CZN 0081
      ICZONE(IPNTR+7)=ICZONE(IPNTR+6)+1                                  CZN 0082
      IPNTR=IPNTR-8                                                      CZN 0083
   30 LNODE=LNODE-1                                                      CZN 0084
      IDOF(1)=CNODE-1+2*NDW                                              CZN 0085
      IDOF(2)=CNODE-1                                                    CZN 0086
      IDOF(3)=CNODE-1+NDW                                                CZN 0087
      IDOF(4)=CNODE+NDW                                                  CZN 0088
      IDOF(5)=CNODE+1+NDW                                                CZN 0089
      IDOF(6)=CNODE+1                                                    CZN 0090
      IDOF(7)=CNODE+1+2*NDW                                              CZN 0091
      IDOF(8)=CNODF+2*NDW                                                CZN 0092
      IPNTR=IMASTR+NET+16*NELW                                           CZN 0093
      ICZONE(IMASTR+NET-1)=IPNTR                                         CZN 0094
      INODE=1                                                            CZN 0095
      DO 40 I=1,8                                                        CZN 0096
      ICZONE(IPNTR+INODE)=IDOF(I)*2                                      CZN 0097
      ICZONE(IPNTR+INODE-1)=ICZONE(IPNTR+INODE)-1                        CZN 0098
   40 INODE=INODE+2                                                      CZN 0099
      IPNTR=IPNTR+16                                                     CZN 0100
      INODE=2*(3*NDW+1)-1                                                CZN 0101
      LNODE=INODE+47                                                     CZN 0102
      DO 50 I=INODE,LNODE                                                CZN 0103
      ICZONE(IPNTR)=I                                                    CZN 0104
   50 IPNTR=IPNTR+1                                                      CZN 0105
      IF(KT1 .EQ. KW) GO TO 80                                           CZN 0106
      WRITE(KW,6002)                                                     CZN 0107
```

```
6002 FORMAT(22H0MASTER ASSEMBLY LIST:,//,18H ELEMENT POINTERS:,/)      CZN 0108
     IMASTR=IBEGIN(4)                                                 CZN 0109
     LMASTR=IMASTR+NET-1                                              CZN 0110
     ELNO=0                                                           CZN 0111
     DO 60 I=IMASTR,LMASTR                                            CZN 0112
     ELNO=ELNO+1                                                      CZN 0113
60   WRITE(KW,6003) ELNO,ICZONE(I)                                    CZN 0114
6003 FORMAT(13H ELEMENT NO.=,I5,13H     POINTER=,I5)                  CZN 0115
     IMASTR=LMASTR+1                                                  CZN 0116
     WRITE(KW,6004)                                                   CZN 0117
6004 FORMAT(17H0ELEMENT D.O.F.S:,/)                                   CZN 0118
     IPNTR=NET-1                                                      CZN 0119
     DO 70 I=1,IPNTR                                                  CZN 0120
     LMASTR=IMASTR+7                                                  CZN 0121
     WRITE(KW,6005) I,(ICZONE(J),J=IMASTR,LMASTR)                     CZN 0122
6005 FORMAT(12H ELEMENT NO.,I5,13H     D.O.F.S:,8I5)                  CZN 0123
70   IMASTR=IMASTR+8                                                  CZN 0124
     LMASTR=IEND(4)                                                   CZN 0125
     IMASTR=LMASTR-63                                                 CZN 0126
     WRITE(KW,6006) (ICZONE(J),J=IMASTR,LMASTR)                       CZN 0127
6006 FORMAT(22H0HOLE ELEMENT D.O.F.S:,/,8(1H ,21X,8I5,/))             CZN 0128
80   CALL ORK(NSIZE,RCZONE,ICZONE)                                    CZN 0129
     CALL QUAD4(COORD,THK,TEMP,C,0,ND,ANGLE,ELQ,ELK,B,1,KW)           CZN 0130
     RATIO=STFTHK/THK                                                 CZN 0131
     DO 90 I=1,36                                                     CZN 0132
90   ELKSTF(I)=ELK(I)*RATIO                                           CZN 0133
     DO 100 I=1,3                                                     CZN 0134
     DO 100 J=1,9                                                     CZN 0135
100  BSTF(I,J)=B(I,J)*RATIO                                           CZN 0136
     IPNTR=NET-1                                                      CZN 0137
     DO 140 N=1,IPNTR                                                 CZN 0138
     DO 110 I=1,4                                                     CZN 0139
     IF( N .EQ. NOTASM(I)) GO TO 140                                  CZN 0140
110  CONTINUE                                                         CZN 0141
     DO 120 I=1,4                                                     CZN 0142
     IF( N .EQ. IELSTF(I)) GO TO 130                                  CZN 0143
```

```
120   CONTINUE
      CALL ASMLTV(N,8,ELK,ELQ,RCZONE,ICZONE)
      GO TO 140
130   CALL ASMLTV(N,8,ELKSTF,ELQ,RCZONE,ICZONE)
140   CONTINUE
      DO 150 I=1,12
150   COORD(I)=0.0
      COORD(1)=2.0*W
      COORD(4)=COORD(1)
      COORD(5)=COORD(1)
      COORD(6)=COORD(1)
      IMASTR=KT1
      KT1=KW
      CALL HOLEL(COORD,THK,S,PI,RHOLE,IHOLE,BHOLE)
      KT1=IMASTR
      CALL ASMSUB(NET,RHOLE,IHOLE,RCZONE,ICZONE)
      ICZONE(1)=CNODE*2
      ICZONE(2)=ICZONE(1)-1
      IMASTR=IBEGIN(5)-1
      DO 160 I=1,2
      J=IMASTR+ICZONE(I)
160   RCZONE(J)=0.0
      CALL BCON(RCZONE,ICZONE)
      I=1
      CALL STACON(I,NIDSS,RCZONE,ICZONE)
      DO 170 I=1,6
170   IBGSS(I)=IBEGIN(I)
      ISZ(1)=NET
      ISZ(2)=NDT
      ISZ(3)=NIDSS
      IF( KT1 .EQ. KW ) GO TO 180
      WRITE(KW,6007)
6007  FORMAT(1H0,59X,10HEXIT CZONE,//)
180   CONTINUE
      RETURN
      END
```

```
      SUBROUTINE FARFLD(WIDTH,LENGTH,NELW,THK,STFTHK,C,S,CNODE,RI,PRESS,
     *INDCTR,NSIZE,RSS,ISS,NSZWRK,RWORK,IWORK)
C***********************************************************************
C SUBROUTINE FARFLD GENERATES A FINITE ELEMENT MODEL ( A SUBSTRUCTURE)
C OF A PLATE WITH ONE HOLE  THAT IS CENTERED IN THE VERTICAL DIRECTION
C AND IS EITHER CENTERED OR OFF-CENTERED IN THE HORIZONTAL DIRECTION
C***********************************************************************
C COPYRIGHT (C) 1975 MASSACHUSETTS INSTITUTE OF TECHNOLOGY
C AEROELASTIC AND STRUCTURES RESEARCH LABORATORY
C***********************************************************************
C FINITE ELEMENT APPLICATIONS TO USAF STRUCTURAL INTEGRITY PROBLEMS
      DIMENSION C(3,3),S(3,3),RSS(1),ISS(1),RWORK(1),IWORK(1),RHOLE(2097
     *),IHOLE(2097),ISAVE1(2),ISAVE2(6),ISAVE3(6),IBGSSC(6),ISZSSC(3),
     *ISZSS(3),ISZ(2)
      INTEGER CNODE
      REAL LENGTH,LNGTH
      COMMON/IO/KR,KW,KP,KT1,KT2,KT3
      COMMON/SIZE/NET,NDT
      COMMON/BEGIN/IBEGIN(6)
      COMMON/END/IEND(6)
      COMMON/SIZESS/NETSS,NDTSS,NIDSS
      COMMON/BEGSS/IBGSS(6)
      EQUIVALENCE (RHOLE(1),IHOLE(1)),(ISZSS(1),NETSS),(ISZ(1),NET)
      NET=3
      NDW=NELW+1
      NDT=8*NDW+48
      NIDSS=NDT-48
      IF(INDCTR .EQ. 1 ) NIDSS=NIDSS-4*NDW
      LMASTR=12*NDW+51
      NCON=  NDW+1
      W=WIDTH/NELW
      LNGTH=LENGTH/2.0-W
      E=WIDTH/2.0-(CNODE-1)*W
      IF(KT1 .EQ. KW) GO TO 10
      WRITE(KW,6000) WIDTH,LENGTH,NELW,THK,STFTHK,CNODE,RI,PRESS,INDCTR
 6000 FORMAT(1H1,59X,12HENTRY FARFLD,///,22H SUBROUTINE INPUT DATA,//,13
```

```
     *H PLATE WIDTH=,E12.5,/,14H PLATE LENGTH=,E12.5,/,33H NUMBER OF ELEFFD 0036
     *MENTS ACROSS WIDTH=,I5,/,17H PLATE THICKNESS=,E12.5,/,21H STIFFENEFFD 0037
     *R THICKNESS=,E12.5,/,33H HOLE CENTER NODE REFERENCE CODE=,I5,/,27HFFD 0038
     * HOLE ELEMENT INPUT RADIUS=,E12.5,/,26H APPLIED TENSION PRESSURE=,FFD 0039
     *E12.5,/,31H STATIC CONDENSATION INDICATOR=,I4,37H    (INDICATOR=0:FFD 0040
     *CHESIRE CAT PROBLEM:,/,39X,73H INDICATOR=1:PLATE WITH END NODES ANFFD 0041
     *D WITH HOLE-NO LOADS OR B.C. APPLIED))                            FFD 0042
      WRITE(KW,6001) LNGTH,F,NIDSS                                      FFD 0043
6001  FORMAT(23H0SUBROUTINE STATISTICS:,//,12H LUG LENGTH=,E12.5,/,13HHOFFD 0044
     *LE OFFSET=,E12.5,/,75H NUMBER OF INTERNAL D.O.F.S TO BE CONDENSED FFD 0045
     *OUT OF GLOBAL STIFFNESS MATRIX=,I5,//)                            FFD 0046
10    CALL SETUP(NSIZE,NCON,LMASTR,RSS,ISS)                             FFD 0047
      IMASTR=IBEGIN(4)                                                  FFD 0048
      IDOF1=1                                                           FFD 0049
      IDOF2=2*(3*NDW+1)-1                                               FFD 0050
      DO 30 N=1,2                                                       FFD 0051
      IPNTR=IMASTR+NET+(N-1)*4*NDW                                      FFD 0052
      ISS(IMASTR+N-1)=IPNTR                                             FFD 0053
      IF( N .EQ. 2) IDOF1=(2*NDW+1)*2-1                                 FFD 0054
      IF( N .EQ. 2) IDOF2=(NDW+1)*2-1                                   FFD 0055
      LMASTR=IDOF1+2*NDW-1                                              FFD 0056
      DO 20 I=IDOF1,LMASTR                                              FFD 0057
      ISS(IPNTR)=I                                                      FFD 0058
20    IPNTR=IPNTR+1                                                     FFD 0059
      LMASTR=IDOF2+2*NDW-1                                              FFD 0060
      DO 30 I=IDOF2,LMASTR                                              FFD 0061
      ISS(IPNTR)=I                                                      FFD 0062
30    IPNTR=IPNTR+1                                                     FFD 0063
      ISS(IMASTR+2)=IPNTR                                               FFD 0064
      IDOF1=1                                                           FFD 0065
      IDOF2=(NDW+1)*2-1                                                 FFD 0066
      LMASTR=NDW*2                                                      FFD 0067
      DO 40 I=IDOF1,LMASTR                                              FFD 0068
      ISS(IPNTR)=I                                                      FFD 0069
40    IPNTR=IPNTR+1                                                     FFD 0070
      LMASTR=4*NDW                                                      FFD 0071
```

```
      DO 50 I=IDOF2,LMASTR                                              FFD 0072
      ISS(IPNTR)=I                                                      FFD 0073
50    IPNTR=IPNTR+1                                                     FFD 0074
      IDOF1=8*NDW+1                                                     FFD 0075
      IDOF2=IDOF1+47                                                    FFD 0076
      DO 60 I=IDOF1,IDOF2                                               FFD 0077
      ISS(IPNTR)=I                                                      FFD 0078
60    IPNTR=IPNTR+1                                                     FFD 0079
      IF(KT1 .EQ. KW) GO TO 80                                          FFD 0080
      IDOF1=IBEGIN(4)                                                   FFD 0081
      IDOF2=IDOF1+2                                                     FFD 0082
      WRITE(KW,6002) (ISS(I),I=IDOF1,IDOF2)                             FFD 0083
6002  FORMAT(22H0MASTER ASSEMBLY LIST:,/,10H POINTERS:,3I5)             FFD 0084
      DO 70 N=1,2                                                       FFD 0085
      IDOF1=IBEGIN(4)+NET+(N-1)*4*NDW                                   FFD 0086
      IDOF2=IDOF1+4*NDW-1                                               FFD 0087
      WRITE(KW,6003) N                                                  FFD 0088
6003  FORMAT(13H0LUG ELEMENT ,I2,9H D.O.F.S:,/)                         FFD 0089
70    WRITE(KW,6004) (ISS(I),I=IDOF1,IDOF2)                             FFD 0090
6004  FORMAT(1H ,20I6)                                                  FFD 0091
      IDOF1=IDOF2+1                                                     FFD 0092
      IDOF2=IEND(4)                                                     FFD 0093
      WRITE(KW,6005)                                                    FFD 0094
6005  FORMAT(23H0CZONE ELEMENT D.O.F.S:,/)                              FFD 0095
      WRITE(KW,6004) (ISS(I),I=IDOF1,IDOF2)                             FFD 0096
80    CALL ORK(NSIZE,RSS,ISS)                                           FFD 0097
      NIDSV=NIDSS                                                       FFD 0098
      DO 90 I=1,2                                                       FFD 0099
90    ISAVE1(I)=ISZ(I)                                                  FFD 0100
      DO 100 I=1,6                                                      FFD 0101
      ISAVE2(I)=IBEGIN(I)                                               FFD 0102
100   ISAVE3(I)=IEND(I)                                                 FFD 0103
      CALL LUG(WIDTH,LNGTH,NELW,THK,STFTHK,C,NSZWRK,RWORK,IWORK)        FFD 0104
      DO 110 I=1,2                                                      FFD 0105
110   ISZ(I)=ISAVE1(I)                                                  FFD 0106
      DO 120 I=1,6                                                      FFD 0107
```

```
      IBEGIN(I)=ISAVE2(I)                                               FFD 0108
  120 IEND(I)=ISAVE3(I)                                                 FFD 0109
      DO 130 N=1,2                                                      FFD 0110
  130 CALL ASMSUB(N,RWORK,IWORK,RSS,ISS)                                FFD 0111
      DO 140 I=1,2                                                      FFD 0112
  140 ISAVE1(I)=ISZ(I)                                                  FFD 0113
      DO 150 I=1,6                                                      FFD 0114
      ISAVE2(I)=IBEGIN(I)                                               FFD 0115
  150 ISAVE3(I)=IEND(I)                                                 FFD 0116
      CALL CZONE(WIDTH,NELW,THK,STFTHK,C,S,CNODE,RI,NSZWRK,RWORK,IWORK, FFD 0117
     $RHOLE,IHOLE)                                                      FFD 0118
      DO 160 I=1,2                                                      FFD 0119
  160 ISZ(I)=ISAVE1(I)                                                  FFD 0120
      DO 170 I=1,6                                                      FFD 0121
      IBEGIN(I)=ISAVE2(I)                                               FFD 0122
      IEND(I)=ISAVE3(I)                                                 FFD 0123
  170 IBGSSC(I)=IBGSS(I)                                                FFD 0124
      DO 180 I=1,3                                                      FFD 0125
  180 ISZSSC(I)=ISZSS(I)                                                FFD 0126
      CALL ASMSUB(3,RWORK,IWORK,RSS,ISS)                                FFD 0127
      NIDSS=NIDSV                                                       FFD 0128
      IF(INOCTR .EQ. 0) GO TO 190                                       FFD 0129
C  PLATE WITH END NODES AND HOLE-NO B.C. OR APPLIED LOADS               FFD 0130
      I=1                                                               FFD 0131
      CALL STACON(I,NIDSS,RSS,ISS)                                      FFD 0132
      GO TO 220                                                         FFD 0133
C  CHESIRE CAT PROBLEM                                                  FFD 0134
C  APPLY B.C.                                                           FFD 0135
  190 IDOF1=(2*NDW+1)*2                                                 FFD 0136
      IDOF2=6*NDW                                                       FFD 0137
      IMASTR=1                                                          FFD 0138
      LMASTR=IBEGIN(5)-1                                                FFD 0139
      DO 200 I=IDOF1,IDOF2,2                                            FFD 0140
      ISS(IMASTR)=I                                                     FFD 0141
      RSS(LMASTR+I)=0.0                                                 FFD 0142
  200 IMASTR=IMASTR+1                                                   FFD 0143
```

```
      IDOF1=(2*NDW+1+NDW/2)*2-1                                         FFD 0144
      ISS(IMASTR)=IDOF1                                                 FFD 0145
      RSS(LMASTR+IDOF1)=0.0                                             FFD 0146
      IDOF1=6*NDW+2                                                     FFD 0147
      IDOF2=8*NDW-2                                                     FFD 0148
      F=     W*PRESS*THK/2.0                                            FFD 0149
      DO 210 I=IDOF1,IDOF2,2                                            FFD 0150
      RSS(LMASTR+I)=RSS(LMASTR+I)+F                                     FFD 0151
  210 RSS(LMASTR+I+2)=RSS(LMASTR+I+2)+F                                 FFD 0152
      F=(STFTHK-THK)     *W*PWFSS/2.0                                   FFD 0153
      RSS(LMASTR+IDOF1  )=RSS(LMASTR+IDOF1  )+F                         FFD 0154
      RSS(LMASTR+IDOF1+2)=RSS(LMASTR+IDOF1+2)+F                         FFD 0155
      RSS(LMASTR+IDOF2+2)=RSS(LMASTR+IDOF2+2)+F                         FFD 0156
      RSS(LMASTR+IDOF2  )=RSS(LMASTR+IDOF2  )+F                         FFD 0157
      CALL BCON(RSS,ISS)                                                FFD 0158
      I=1                                                               FFD 0159
      CALL  STACON(I,NIDSS,RSS,ISS)                                     FFD 0160
  220 DO 221 I=1,2                                                      FFD 0161
  221 ISZSS(I)=ISZ(I)                                                   FFD 0162
      ISZSS(3)=NIDSS                                                    FFD 0163
      DO 222 I=1,6                                                      FFD 0164
  222 IBGSS(I)=IBEGIN(I)                                                FFD 0165
      IF( KTI .EQ. KW ) GO TO 230                                       FFD 0166
      WRITE(KW,6006)                                                    FFD 0167
 6006 FORMAT(1H0,59X,11HEXIT FARFLD,///)                                FFD 0168
  230 CONTINUE                                                          FFD 0169
      RETURN                                                            FFD 0170
      END                                                               FFD 0171
```

```
      SUBROUTINE ARC4(RINNER,THETA1,DTHT,C,THK,ELK1,RSS,ISS,ISIZE,IBEGINARC40000
     *,B)                                                               ARC40001
C***********************************************************************ARC40002
C SUBROUTINE ARC4 SUPPLIES THE 150 DEGREE ARC SUBSTRUCTURE USED IN      ARC40003
C SUBROUTINE RING                                                       ARC40004
C***********************************************************************ARC40005
C AEROELASTIC AND STRUCTURES RESEARCH LABORATORY                        ARC40006
C COPYRIGHT (C) 1975 MASSACHUSETTS INSTITUTE OF TECHNOLOGY              ARC40007
C***********************************************************************ARC40008
C FINITE ELEMENT APPLICATIONS TO USAF STRUCTURAL INTEGRITY PROBLEMS     ARC40009
      DIMENSION C(3,3),ELK1(8,8),                                       ARC40010
     *ELK (36),RSS(4504),ISS(4504 ),ISIZE(3),IBEGIN(6),                 ARC40011
     *NODE(4),COORD4(12),                   ISAVE1(2),ISAVE2(6),ISAVE3(6ARC40012
     *),TEMP4(4),ND4(4),ANGLE4(4)                                       ARC40013
      DIMENSION FLO(8),B(3,8)                                           ARC40014
      DATA COORD4/12*0.0/,TEMP4/4*0.0/                                  ARC40015
      COMMON/IO/KR,KW,KP,KT1,KT2,KT3                                    ARC40016
      COMMON/SIZE/ISZ(2)                                                ARC40017
      COMMON/BEGIN/IBG(6)                                               ARC40018
      COMMON/END/IND(6)                                                 ARC40019
C  SAVE CALLING PROGRAM CONTROLS                                        ARC40020
      ISAVE1(1)=ISZ(1)                                                  ARC40021
      ISAVE1(2)=ISZ(2)                                                  ARC40022
      DO 10 I=1,6                                                       ARC40023
      ISAVE2(I)=IBG(I)                                                  ARC40024
   10 ISAVE3(I)=IND(I)                                                  ARC40025
      ISZ(1)=40                                                         ARC40026
      ISZ(2)=110                                                        ARC40027
      NID=54                                                            ARC40028
      NSIZE=4504                                                        ARC40029
      CALL SETUP(NSIZE,1,360,RSS,ISS)                                   ARC40030
C  ESTABLISH ASSEMBLY LIST                                              ARC40031
      IMASTR=IBG(4)                                                     ARC40032
      LMASTR=IND(4)                                                     ARC40033
      NODE(1)=55                                                        ARC40034
      NODE(2)=54                                                        ARC40035
```

```
      DO 60 N=1,10                                                      ARC40036
      NM1=N-1                                                           ARC40037
      IPNTR=IMASTR+ISZ(1)+NM1*8                                         ARC40038
      IDMMY=IMASTR+NM1                                                  ARC40039
      ISS(IDMMY)=IPNTR                                                  ARC40040
      IF(N .NE. 1) GO TO 20                                             ARC40041
      NODE(3)=27                                                        ARC40042
      NODE(4)=28                                                        ARC40043
      GO TO 40                                                          ARC40044
20    IF( N .NE. 10 ) GO TO 30                                          ARC40045
      NODE(3)=38                                                        ARC40046
      NODE(4)=37                                                        ARC40047
      GO TO 40                                                          ARC40048
30    NODE(3)=NODE(2)-3                                                 ARC40049
      NODE(4)=NODE(1)+1                                                 ARC40050
40    CONTINUE                                                          ARC40051
      DO 50 I=1,4                                                       ARC40052
      IADD=IPNTR+I*2-1                                                  ARC40053
      IDOF=NODE(I)*2                                                    ARC40054
      ISS(IADD-1)=IDOF-1                                                ARC40055
50    ISS(IADD)=IDOF                                                    ARC40056
      NODE(1)=NODE(4)                                                   ARC40057
60    NODE(2)=NODE(3)                                                   ARC40058
      NODE(1)=54                                                        ARC40059
      NODE(2)=53                                                        ARC40060
      DO 110 N=11,20                                                    ARC40061
      IPNTR=IPNTR+8                                                     ARC40062
      IDMMY=IMASTR+N-1                                                  ARC40063
      ISS(IDMMY)=IPNTR                                                  ARC40064
      IF( N .NE. 11 ) GO TO 70                                          ARC40065
      NODE(3)=26                                                        ARC40066
      NODE(4)=27                                                        ARC40067
      GO TO 90                                                          ARC40068
70    IF( N .NE. 20 ) GO TO 80                                          ARC40069
      NODE(3)=39                                                        ARC40070
      NODE(4)=38                                                        ARC40071
```

```
      GO TO 90                                                   ARC40072
   80 NODE(3)=NODE(2)-3                                          ARC40073
      NODE(4)=NODE(1)-3                                          ARC40074
   90 CONTINUE                                                   ARC40075
      DO 100 I=1,4                                               ARC40076
      IADD=IPNTR+I*2-1                                           ARC40077
      IDOF=NODE(I)*2                                             ARC40078
      ISS(IADD-1)=IDOF-1                                         ARC40079
  100 ISS(IADD)=IDOF                                             ARC40080
      NODE(1)=NODE(4)                                            ARC40081
  110 NODE(2)=NODE(3)                                            ARC40082
      NODE(1)=53                                                 ARC40083
      NODE(2)=52                                                 ARC40084
      DO 160 N=21,30                                             ARC40085
      IPNTR=IPNTR+8                                              ARC40086
      IDMMY=IMASTR+N-1                                           ARC40087
      ISS(IDMMY)=IPNTR                                           ARC40088
      IF( N .NE. 21 ) GO TO 120                                  ARC40089
      NODE(3)=25                                                 ARC40090
      NODE(4)=26                                                 ARC40091
      GO TO 140                                                  ARC40092
  120 IF( N .NE. 30 ) GO TO 130                                  ARC40093
      NODE(3)=40                                                 ARC40094
      NODE(4)=39                                                 ARC40095
      GO TO 140                                                  ARC40096
  130 NODE(3)=NODE(2)-3                                          ARC40097
      NODE(4)=NODE(1)-3                                          ARC40098
  140 CONTINUE                                                   ARC40099
      DO 150 I=1,4                                               ARC40100
      IADD=IPNTR+I*2-1                                           ARC40101
      IDOF=NODE(I)*2                                             ARC40102
      ISS(IADD-1)=IDOF-1                                         ARC40103
  150 ISS(IADD)=IDOF                                             ARC40104
      NODE(1)=NODE(4)                                            ARC40105
  160 NODE(2)=NODE(3)                                            ARC40106
      NODE(1)=52                                                 ARC40107
```

```
      NODE(2)=51                                                        ARC40108
      DO 169 N=31,40                                                    ARC40109
      IPNTR=IPNTR+8                                                     ARC40110
      IDMMY=IMASTR+N-1                                                  ARC40111
      ISS(IDMMY)=IPNTR                                                  ARC40112
      IF( N .NE. 31 ) GO TO 161                                         ARC40113
      NODE(3)=50                                                        ARC40114
      NODE(4)=25                                                        ARC40115
      GO TO 162                                                         ARC40116
  161 IF( N .NE. 40 ) GO TO 1615                                        ARC40117
      NODE(3)=41                                                        ARC40118
      NODE(4)=40                                                        ARC40119
      GO TO 162                                                         ARC40120
 1615 NODE(3)=NODE(2)-1                                                 ARC40121
      NODE(4)=NODE(1)-3                                                 ARC40122
  162 CONTINUE                                                          ARC40123
      DO 163 I=1,4                                                      ARC40124
      IADD=IPNTR+I*2-1                                                  ARC40125
      IDOF=NODE(I)*2                                                    ARC40126
      ISS(IADD-1)=IDOF-1                                                ARC40127
  163 ISS(IADD)=IDOF                                                    ARC40128
      NODE(1)=NODE(4)                                                   ARC40129
  169 NODE(2)=NODE(3)                                                   ARC40130
      CALL OPK(NSIZE,RSS,ISS)                                           ARC40131
C  OBTAIN ARC ELEMENTS:                                                 ARC40132
      RI=RINNER                                                         ARC4..133
      RO=RI+RI*DTHT                                                     ARC40134
      CS=COS(DTHT)                                                      ARC40135
      SS=SIN(DTHT)                                                      ARC40136
C  INNER RING ELEMENT:                                                  ARC40137
      COORD4(1)=RI                                                      ARC40138
      COORD4(4)=RO                                                      ARC40139
      COORD4(7)=RO*CS                                                   ARC40140
      COORD4(8)=RO*SS                                                   ARC40141
      COORD4(10)=RI*CS                                                  ARC40142
      COORD4(11)=RI*SS                                                  ARC40143
```

```
      CALL QUAD4(COORD4,THK,TEMP4,C,0,ND4,ANGLE4,ELQ, ELK ,8 ,1,KW)     ARC40144
      L=0                                                               ARC40145
      DO 170 I=1,8                                                      ARC40146
      DO 170 J=1,I                                                      ARC40147
      L=L+1                                                             ARC40148
      ELK1(I,J)=ELK (L)                                                 ARC40149
  170 ELK1(J,I)=ELK1(I,J)                                               ARC40150
C  MID RING & OUTER RING ELEMENTS ARE OF SIMILAR GEOMETRY AND HAVE SAME ARC40151
C  STIFFNESS MATRICES                                                   ARC40152
C  ROTATE ELEMENTS AROUND ARC                                           ARC40153
      THETA=THETA1-DTHT                                                 ARC40154
      NI=0                                                              ARC40155
      NM1=10                                                            ARC40156
      NM2=20                                                            ARC40157
      NO=30                                                             ARC40158
      DO 200 I=1,10                                                     ARC40159
      THETA=THETA+DTHT                                                  ARC40160
      NI=NI+1                                                           ARC40161
      NM1=NM1+1                                                         ARC40162
      NM2=NM2+1                                                         ARC40163
      NO=NO+1                                                           ARC40164
      CALL TRANS(THETA,ELK1,ELK,8)                                      ARC40165
      CALL ASMLTV(NI ,8,ELK,ELQ,RSS,ISS)                                ARC40166
      CALL ASMLTV(NM1,8,ELK,ELQ,RSS,ISS)                                ARC40167
      CALL ASMLTV(NM2,8,ELK,ELQ,RSS,ISS)                                ARC40168
      CALL ASMLTV(NO ,8,ELK,ELQ,RSS,ISS)                                ARC40169
  200 CONTINUE                                                          ARC40170
C  STATIC CONDENSATION                                                  ARC40171
      ISIGN=1                                                           ARC40172
      CALL STACON(ISIGN,NID,RSS,ISS)                                    ARC40173
C  RETURN CALLING PROGRAM CONTROLS TO THEIR ORIGINAL STORAGE            ARC40174
      ISIZE(1)=ISZ(1)                                                   ARC40175
      ISIZE(2)=ISZ(2)                                                   ARC40176
      ISIZE(3)=NID                                                      ARC40177
      DO 300 I=1,6                                                      ARC40178
      IBEGIN(I)=IBG(I)                                                  ARC40179
```

```
      IBG(I)=ISAVE2(I)                                                  ARC40180
  300 IND(I)=ISAVE3(I)                                                  ARC40181
      ISZ(1)=ISAVE1(1)                                                  ARC40182
      ISZ(2)=ISAVE1(2)                                                  ARC40183
      RETURN                                                            ARC40184
      END                                                               ARC40185
```

```
      SUBROUTINE TRANS(THETA,ELKSQ,ELKV,IDOF)
C***********************************************************************
C SUBROUTINE TRANS TRANSFORMS QUAD4 ELEMENTS AROUND THE 150 DEGREE ARC
C MESH OF SUBROUTINE ARC4 AND ALONG THE CRACK RAYS OF SUBROUTINF RING
C***********************************************************************
C COPYRIGHT (C) 1975 MASSACHUSETTS INSTITUTE OF TECHNOLOGY
C AEROELASTIC AND STRUCTURES RESEARCH LABORATORY
C***********************************************************************
C FINITE ELEMENT APPLICATIONS TO USAF STRUCTURAL INTEGRITY PROBLEMS
      DIMENSION ELKSQ(IDOF,IDOF),ELKV(1),T(8,8)
      DO 5 I=1,IDOF
      DO 5 J=1,IDOF
 5    T(I,J)=0.0
      CS=COS(THETA)
      SS=SIN(THETA)
      T(1,1)=CS
      T(1,2)=SS
      T(2,2)=CS
      T(2,1)=-SS
      DO 10 IT=2,IDOF,2
      ITM2=IT-2
      DO 10 I=1,2
      DO 10 J=1,2
10    T(ITM2+I,ITM2+J)= T(I,J)
      L=0
      DO 20 I=1,IDOF
      DO 20 J=1,I
      L=L+1
      ELKV(L)=0.0
      DO 20 K=1,IDOF
      DO 20 LL=1,IDOF
20    ELKV(L)=ELKV(L)+T(K,I)*ELKSQ(K,LL)*T(LL,J)
      RETURN
      END
```

```
      SUBROUTINE RING(RINNER,THK,THTCRK,NCRK,A1,A2,INDCTR,S,C,NSIZE,
     *RSS,ISS)
C***********************************************************************
C SUBROUTINE RING GENERATES A FINITE ELEMENT MODEL (A SUBSTRUCTURE)
C OF A 2 DIMENSIONAL CRACKED RING WHICH MAY BE ASSEMBLED INTO A HOLE
C OF ANOTHER FINITE ELEMENT MODEL
C***********************************************************************
C COPYRIGHT (C) 1975 MASSACHUSETTS INSTITUTE OF TECHNOLOGY
C AEROELASTIC AND STRUCTURES RESEARCH LABORATORY
C***********************************************************************
C FINITE ELEMENT APPLICATIONS TO USAF STRUCTURAL INTEGRITY PROBLEMS
      DIMENSION RCRKTP(2),ICASE(2),RSS(1000),ISS(1000)        ,NODEL1(3
     *2),NODEL2(32),NOTASM(20),NDCRK1(9,4),NDCRK2(9,4),S(3,3),C(3,3),ELK
     *(8,8),ELKV(36),ELQ(8),ISAVE1(2),ISAVE2(6),ISAVE3(6),ISVSS1(3),ISVS
     *S2(6),NDRNG1(28),NDRNG2(28)
      DIMENSION RARC(4504),IARC(4504),BARC(3,9)
      DIMENSION EK(171),RCR(2,18)               ,COORD(18),EQ(18)
      EQUIVALENCE(RARC(1),IARC(1))
      COMMON/IO/KR,KW,KP,KT1,KT2,KT3
      COMMON/SIZE/ISZ(2)
      COMMON/BEGIN/IBG(6)
      COMMON/END/IEND(6)
      COMMON/SIZESS/ISZSS(3)
      COMMON/BEGSS/IBEGSS(6)
      COMMON/MTLPRM/SMU,ETA
      COMMON/KI/RCRK(2,2,18)
      COMMON/RADIUS/RNODE(5)
      DATA EQ/18*0.0/
      DATA ELQ/8*0.0/,PI/3.141593/
      DATA NODEL1/47,18,12,48,18,16,10,12,16,14,8,10,14,71,72,8,
     *24,22,6,25,22,20,4,6,20,8,2,4,8,72,49,2/
      DATA NODEL2/35,17,11,36,17,15,9,11,15,13,7,9,13,59,60,7,
     *23,21,5,37,21,19,3,5,19,7,1,3,7,60,61,1/
      DATA NOTASM/20*0/
      DATA NDCRK1/10,4,6,25,24,48,47,18,16,8,2,4,6,22,12,18,16,14,
     *72,49,2,4,20,10,16,14,71,9*20/
```

```
      DATA NDCRK2/9,3,5,37,23,36,35,17,15,7,1,3,5,21,11,17,15,13,        RING0036
     @60,61,1,3,19,9,15,13,59,9*19/                                      RING0037
      DATA NDRNG1/38,39,40,41,42,43,44,45,46,47,18,16,14,                RING0038
     @71,70,69,68,67,66,65,64,63,62,61,1,3,5,37/                         RING0039
      DATA NDRNG2/26,27,28,29,30,31,32,33,34,35,17,15,13,59,             RING0040
     @58,57,56,55,54,53,52,51,50,49,2,4,6,25/                            RING0041
C  SAVE CALLING PROGRAM FEABL CONTROLS                                   RING0042
      DO 101 I=1,6                                                       RING0043
      ISAVE2(I)=IBG(I)                                                   RING0044
 101  ISAVE3(I)=IEND(I)                                                  RING0045
      DO 103 I=1,2                                                       RING0046
 103  ISAVE1(I)=ISZ(I)                                                   RING0047
      RCRKTP(1)=RINNER+A1                                                RING0048
      RCRKTP(2)=RINNER+A2                                                RING0049
      DTHT=PI/12.0                                                       RING0050
      RNODE(1)=RINNER                                                    RING0051
      DO 10 I=1,4                                                        RING0052
 10   RNODE(I+1)=RNODE(I)*(1.0+DTHT)                                     RING0053
      RO=RNODE(5)                                                        RING0054
      NET=20                                                             RING0055
      NDT=144                                                            RING0056
      ISZ(1)=20                                                          RING0057
      ISZ(2)=144                                                         RING0058
      DELTA=(RNODE(5)+RNODE(4))/2.0                                      RING0059
      RNODE(5)=RNODE(4)                                                  RING0060
      RNODE(4)=DELTA                                                     RING0061
      DO 40 I=1,2                                                        RING0062
      IF(RCRKTP(I) .GT. DELTA) GO TO 200                                 RING0063
      IF(RCRKTP(I) .EQ. RINNER) GO TO 30                                 RING0064
      DO 20 J=1,3                                                        RING0065
 20   IF(RCRKTP(I) .GT. RNODE(J) .AND. RCRKTP(I) .LE. RNODE(J+1)) ICASE(RING0066
     @I)=J                                                               RING0067
      GO TO 40                                                           RING0068
 30   ICASE(I)=4                                                         RING0069
 40   CONTINUE                                                           RING0070
      RNODE(4)=RNODE(5)                                                  RING0071
```

```
      RNODE(5)=RO                                                       RING0072
      NIDSS=96                                                          RING0073
      IF(INDCTR .EQ. 1) GO TO 105                                       RING0074
      NIDSS=48-NCRK*2                                                   RING0075
 105  CONTINUE                                                          RING0076
      IF(KT1 .EQ. KW) GO TO 9331                                        RING0077
      WRITE(KW,9000)RINNER,THK,NCRK,THTCRK,A1,A2,INDCTR,(RCRKTP(I),I=1,2RING0078
     &)                                                                 RING0079
 9000 FORMAT(1H1,60X,10HENTRY RING,//,23H SUBROUTINE INPUT DATA:,/,17H RRING0080
     &ADIUS OF HOLE =,E12.5,/,19H THICKNESS OF RING=,E12.5,/,18H NUMBER RING0081
     &OF CRACKS=,I3,/,32H ANGLE OF CRACK NO. 1 (RADIANS)=,E12.5,42H    (RING0082
     &CRACK NO. 2 IS 180 DEGREES OPPOSITE ),/,20H CRACK NO. 1 LENGTH=,E1RING0083
     &2.5,22H   CRACK NO. 2 LENGTH=,E12.5,//,24H CONDENSATION INDICATOR=RING0084
     &,I3,99H     (INDICATOR=0 : D.O.F.S AROUND HOLE REMAIN ;     INDICARING0085
     &TOR=1 : D.O.F.S AROUND HOLE ARE REMOVED),///,23H SUBROUTINE STATISRING0086
     &TICS:,//,52H RADIUS DIMENSION OF CRACK TIPS:     R CRACK NO. 1 =,ERING0087
     &12.5,19H     R CRACK NO. 2=,E12.5)                                RING0088
      WRITE(KW,9001)(RNODE(I),I=1,5)                                    RING0089
 9001 FORMAT(35H INNER RADII OF CONCENTRIC RINGS : ,/,34X,E12.5,/,34X,E1RING0090
     &2.5,/34X,E12.5,/,34X,E12.5,/,22H OUTER RADIUS OF RING=,E12.5)     RING0091
      WRITE(KW,9002)(ICASE(I),I=1,2)                                    RING0092
 9002 FORMAT(26H CASE REFERENCE NUMBERS : ,2I5)                         RING0093
      WRITE(KW,9009)NIDSS                                               RING0094
 9009 FORMAT(71H NO. OF INTERNAL D.O.F.S TO BE CONDENSED OUT OF RING STIRING0095
     &FFNESS MATRIX =,I5)                                               RING0096
 9331 CALL SETUP(NSIZE,14,296,RSS,ISS)                                  RING0097
C  ESTABLISH ASSEMBLY LIST                                              RING0098
      IMASTR=IBS(4)                                                     RING0099
      LMASTR=IEND(4)                                                    RING0100
      IPNTR=IMASTR+NET                                                  RING0101
      ISS(IMASTR)=IPNTR                                                 RING0102
      DO 50 I=1,28                                                      RING0103
      IADD=IPNTR+I*2-2                                                  RING0104
      IDOF=NDRNG1(I)*2                                                  RING0105
      ISS(IADD)=IDOF-1                                                  RING0106
 50   ISS(IADD+1)=IDOF                                                  RING0107
```

```
      IPNTR=IPNTR+56                                                    RING0108
      ISS(IMASTR+1)=IPNTR                                               RING0109
      DO 60 I=1,28                                                      RING0110
      IADD=IPNTR+I*2-2                                                  RING0111
      IDOF=NDRNG2(I)*2                                                  RING0112
      ISS(IADD)=IDOF-1                                                  RING0113
   60 ISS(IADD+1)=IDOF                                                  RING0114
      DO 5 I=1,2                                                        RING0115
      N=ICASE(I)                                                        RING0116
      GO TO (1,2,3,4),N                                                 RING0117
    1 IF(I .EQ. 2) GO TO 12                                             RING0118
      DO 11 N=3,4                                                       RING0119
      NP4=N+4                                                           RING0120
      NOTASM(NP4)=NP4                                                   RING0121
   11 NOTASM(N)=N                                                       RING0122
      NODEL1(22)=10                                                     RING0123
      NODEL1(25)=10                                                     RING0124
      GO TO 5                                                           RING0125
   12 DO 13 N=11,12                                                     RING0126
      NP4=N+4                                                           RING0127
      NOTASM(NP4)=NP4                                                   RING0128
   13 NOTASM(N)=N                                                       RING0129
      NODEL2(22)=9                                                      RING0130
      NODEL2(25)=9                                                      RING0131
      GO TO 5                                                           RING0132
    2 IF(I .EQ. 2) GO TO 22                                             RING0133
      DO 21 N=4,5                                                       RING0134
      NP4=N+4                                                           RING0135
      NOTASM(NP4)=NP4                                                   RING0136
   21 NOTASM(N)=N                                                       RING0137
      GO TO 5                                                           RING0138
   22 DO 23 N=12,13                                                     RING0139
      NP4=N+4                                                           RING0140
      NOTASM(NP4)=NP4                                                   RING0141
   23 NOTASM(N)=N                                                       RING0142
      GO TO 5                                                           RING0143
```

```
 3    IF(I .EQ. 2) GO TO 32                                   RING0144
      DO 31 N=5.6                                             RING0145
      NP4=N+4                                                 RING0146
      NOTASM(NP4)=NP4                                         RING0147
 31   NOTASM(N)=N                                             RING0148
      GO TO 5                                                 RING0149
 32   DO 33 N=13.14                                           RING0150
      NP4=N+4                                                 RING0151
      NOTASM(NP4)=NP4                                         RING0152
 33   NOTASM(N)=N                                             RING0153
      GO TO 5                                                 RING0154
 4    IF(I .EQ. 2) GO TO 41                                   RING0155
      NODEL1(17)=48                                           RING0156
      NODEL1(18)=12                                           RING0157
      NODEL1(21)=12                                           RING0158
      NODEL1(22)=10                                           RING0159
      NODEL1(25)=10                                           RING0160
      NOTASM(19)=19                                           RING0161
      GO TO 5                                                 RING0162
 41   NODEL2(17)=36                                           RING0163
      NODEL2(18)=11                                           RING0164
      NODEL2(21)=11                                           RING0165
      NODEL2(22)=4                                            RING0166
      NODEL2(25)=4                                            RING0167
      NOTASM(20)=20                                           RING0168
 5    CONTINUE                                                RING0169
      DO 70 N=3.10                                            RING0170
      IPNTR=IMASTR+NFT+112+(N-3)*8                            RING0171
      ISS(IMASTR+N-1)=IPNTR                                   RING0172
      DO 70 I=1.4                                             RING0173
      IADD=IPNTR+(I-1)*2                                      RING0174
      ISS(IADD)=NODEL1((N-3)*4+I)*2-1                         RING0175
 70   ISS(IADD+1)=ISS(IADD)+1                                 RING0176
      DO 80 N=11.18                                           RING0177
      IPNTR=IMASTR+NFT+176+(N-11)*8                           RING0178
      ISS(IMASTR+N-1)=IPNTR                                   RING0179
```

```
      DO 80 I=1,4                                                       RING0180
      IADD=IPNTR+(I-1)*2                                                RING0181
      ISS(IADD)=NODEL2((N-11)*4+I)*2-1                                  RING0182
80    ISS(IADD+1)=ISS(IADD)+1                                           RING0183
      DO 100 N=19,20                                                    RING0184
      IPNTR=IMASTR+NET*240 +(N-19)*18                                   RING0185
      ISS(IMASTR+N-1)=IPNTR                                             RING0186
      DO 100 I=1,9                                                      RING0187
      IADD=IPNTR+(I-1)*2                                                RING0188
      IF(N .EQ. 20 ) GO TO 90                                           RING0189
      ISS(IADD)=NDCPK1(I,ICASE(1))*2-1                                  RING0190
      ISS(IADD+1)=ISS(IADD)+1                                           RING0191
      GO TO 100                                                         RING0192
90    ISS(IADD)=NDCPK2(I,ICASE(2))*2-1                                  RING0193
      ISS(IADD+1)=ISS(IADD)+1                                           RING0194
100   CONTINUE                                                          RING0195
      IF(KT1 .EQ. KW) GO TO 1235                                        RING0196
      WRITE(KW,9004)(NOTASM(I),I=1,20)                                  RING0197
9004  FORMAT(24H ELEMENTS NOT ASSEMBLED:,/,20I5)                        RING0198
      WRITE(KW,9003)(ISS(I),I=IMASTR,LMASTR)                            RING0199
9003  FORMAT(22H MASTER ASSEMBLY LIST:,/,10H POINTERS:,20I5,/,32H ELEMENRING0200
     *T D.O.F.S: ELEMENTS 1-18:,/,15(1H0,16I5,/),/,49H ELEMENT D.O.F.S: RING0201
     *ELEMENT D.O.F.S: ELEMENT 19-20:,/,2(1H0,18I5,/))                  RING0202
1235   CONTINUE                                                         RING0203
      CALL ORK(NSIZE,RSS,ISS)                                           RING0204
C  ASSEMBLE ELEMENTS                                                    RING0205
      THETA=THTCRK+PI+DTHT                                              RING0206
      CALL ARC4(RINNER,THETA,DTHT,C,THK,ELK,RARC,IARC,ISZSS,IBEGSS,BARC)RING0207
      DO 120 N=1,2                                                      RING0208
120   CALL ASMSUP(N,RARC,IARC,RSS,ISS)                                  RING0209
      THETA=THTCRK-DTHT                                                 RING0210
      CALL TRANS(THETA,ELK,ELKV,8)                                      RING0211
      DO 130 N=3,6                                                      RING0212
      NP4=N+8                                                           RING0213
      IF(N .EQ. NOTASM(N)) GO TO 125                                    RING0214
      CALL ASMLTV(N,8,ELKV,ELQ,RSS,ISS)                                 RING0215
```

```
  125 CONTINUE                                                    RING0216
      IF(NP4 .EQ. NOTASM(NP4)) GO TO 130                          RING0217
      CALL ASMLTV(NP4,R,ELKV,FLO,RSS,ISS)                         RING0218
  130 CONTINUE                                                    RING0219
      THETA=THTCRK                                                RING0220
      CALL TRANS(THETA,ELK,ELKV,R)                                RING0221
      DO 140 N=7,10                                               RING0222
      NP4=N+8                                                     RING0223
      IF(N .EQ. NOTASM(N)) GO TO 135                              RING0224
      CALL ASMLTV(N,R,ELKV,FLO,RSS,ISS)                           RING0225
  135 CONTINUE                                                    RING0226
      IF(NP4 .EQ. NOTASM(NP4)) GO TO 140                          RING0227
      CALL ASMLTV(NP4,R,ELKV,FLO,RSS,ISS)                         RING0228
  140 CONTINUE                                                    RING0229
      DO 160 N=19,20                                              RING0230
      NM18=N-18                                                   RING0231
      IF(N.EQ. NOTASM(N)) GO TO 156                               RING0232
C   PCRK59 COORDINATES                                            RING0233
      CS1=COS(THETA-DTHT)                                         RING0234
      SS1=SIN(THETA-DTHT)                                         RING0235
      CS2=COS(THETA)                                              RING0236
      SS2=SIN(THETA)                                              RING0237
      CS3=COS(THETA+DTHT)                                         RING0238
      SS3=SIN(THETA+DTHT)                                         RING0239
      XT=RCRKTP(NM18)*CS2                                         RING0240
      YT=RCRKTP(NM18)*SS2                                         RING0241
      RO=RNODE(ICASE(NM18),2)                                     RING0242
      COORD(1)=CS2*RO                                             RING0243
      COORD(2)=SS2*RO                                             RING0244
      DO 150 I=1,3                                                RING0245
      J=I*2+1                                                     RING0246
      RO=RNODE(ICASE(NM18),3-I)                                   RING0247
      COORD(J)=RO*CS3                                             RING0248
      COORD(J+1)=RO*SS3                                           RING0249
      J=(I+5)*2+1                                                 RING0250
      RO=RNODE(ICASE(NM18),I-1)                                   RING0251
```

```
      COORD(J)=RO*CS1                                                   RING0252
  150 COORD(J+1)=RO*SS1                                                 RING0253
      RO=RNODE(ICASE(NM18))                                             RING0254
      COORD(9)=RO*CS2                                                   RING0255
      COORD(10)=RO*SS2                                                  RING0256
      COORD(11)=COORD( 9)                                               RING0257
      COORD(12)=COORD(10)                                               RING0258
      CALL PCRK59(2,SMU,ETA,XT,YT, COORD,EK,BCR)                        RING0259
      DO 155 I=1,2                                                      RING0260
      DO 155 J=1,18                                                     RING0261
  155 BCRK(NM18,I,J)=BCR(I,J)                                           RING0262
      CALL ASMLTV(N,18,EK,EO,RSS,ISS)                                   RING0263
      GO TO 160                                                         RING0264
  156 CONTINUE                                                          RING0265
      DO 157 I=1,2                                                      RING0266
      DO 157 J=1,18                                                     RING0267
  157 BCRK(NM18,I,J)=0.0                                                RING0268
  160 THETA=THTCRK*PI                                                   RING0269
C  DETERMINE CONSTRAINED D.O.F.S                                        RING0270
      ICN=0                                                             RING0271
      IDIAG=0                                                           RING0272
      DO 190 I=1,144                                                    RING0273
      IDIAG=IDIAG+1                                                     RING0274
      J=ISS(IBG(2)+IDIAG-1)+IDIAG                                       RING0275
      IF(RSS(J) .NE. 0.0) GO TO 190                                     RING0276
      ICN=ICN+1                                                         RING0277
      ISS(ICN)=IDIAG                                                    RING0278
      RSS(IBG(5)    -1+IDIAG)=0.0                                       RING0279
  190 CONTINUE                                                          RING0280
      CALL BCON(RSS,ISS)                                                RING0281
      I=1                                                               RING0282
      CALL STACON(I,NIDSS,RSS,ISS)                                      RING0283
      RO=RNODE(5)                                                       RING0284
      DO 195 I=1,6                                                      RING0285
      IBEGSS(I)=IBG(I)                                                  RING0286
      IBG(I)=ISAVF2(I)                                                  RING0287
```

```
195   IEND(I)=ISAVE3(I)
      DO 196 I=1,2                                                      RING0288
      ISZSS(I)=ISZ(I)                                                   RING0289
196   ISZ(I)=ISAVF1(I)                                                  RING0290
      ISZSS(3)=NIDSS                                                    RING0291
      IF(KT1 .EQ. KW) GO TO 197                                         RING0292
      WRITE(KW,1234)                                                    RING0293
1234  FORMAT(1H0,60X,9HEXIT RING)                                       RING0294
197   RETURN                                                            RING0295
200   WRITE(KW,6000)I,RCRKTP(I),RNODE(4)                                RING0296
6000  FORMAT(14HOLENGTH OF NO.,I2,10H CRACK IS ,E12.5,81H MEASURED FROM RING0297
     *CENTER OF HOLE AND IS GREATER THAN MAXIMUM CRACK LENGTH ALLOWED BYRING0298
     *,/,17H THIS CODEING OF ,E12.5,42H. EXECUTION TERMINATED IN SURROUTRING0299
     *INE RING.)                                                        RING0300
      STOP                                                              RING030
      END                                                               RING0302
                                                                        RING0303
```

```
C  TEST MAIN - CONTRACT 81739  PROBLEM 3  (LEFT SIDE STIFFENED )
C***********************************************************************
C COPYRIGHT (C) 1975 MASSACHUSETTS INSTITUTE OF TECHNOLOGY
C AEROELASTIC AND STRUCTURES RESEARCH LABORATORY
C***********************************************************************
C FINITE ELEMENT APPLICATIONS TO USAF STRUCTURAL INTEGRITY PROBLEMS
C........INPUT DATA FOR PROBLEM #3......................................
C SET        VARIABLES IN SET                        FORMAT FOR EACH CARD
C  1         WIDTH,LENGTH,THK,STFFCT,PRESS,RI,IOFFST           (6E10.3,I5)
C  2         A(1),A(2),IPOS(1),IPOS(2)                        (2E10.3,2I5)
C  3         E,GNU                                                (2E10.3)
C  4         KT1,KT2,KT3                                             (3I5)
C =====DEFINITIONS OF INPUT VARIABLES=====================================
C WIDTH = WIDTH OF PLATE;  LENGTH = LENGTH OF PLATE;
C THK = THICKNESS OF PLATE;  STFFCT = STIFFNESS FACTOR;
C PRESS = FARFIELD LOADING;  RI = RADIUS OF HOLE;
C IOFFSET = OFFSET INDICATOR;  A(1) = LENGTH OF CRACK ONE;
C A(2) = LENGTH OF CRACK TWO; IPOS(1) = INITIAL POSITION OF CRACK ONE;
C IPOS(2) = FINAL POSITION OF CRACK ONE;  E = YOUNGS MODULUS;
C GNU = POISSONS RATIO;KT1 = PRINT CONTROL FOR FEABL OPTIONAL PRINT;
C KT2 = PRINT CONTROL FOR OPTIONAL FARFLD PRINTING;
C KT3 = PRINT CONTROL FOR OPTIONAL RING PRINTING;
      DIMENSION RFF(10000),IFF(10000),REAL(8000),INTGR(8000),RRNG(12000)
     @,IRNG(12000),S(3,3),C(3,3),IBEGIN(6),IEND(6),ISZ(2),ISZSS(3),
     @ANGLE(2,2),A(2),IPOS(2),ISZFF(3),IBGFF(6),ISAVE(6),ELQ(18),RK(2)
      REAL LENGTH
      INTEGER CNODE,PRNTSV
      COMMON/IO/KR,KW,KP,KT1,KT2,KT3
      COMMON/SIZE/NET,NDT
      COMMON/BEGIN/ICON,IKOUNT,ILNZ,IMASTR,IQ,IK
      COMMON/END/LCON,LKOUNT,LLNZ,LMASTR,LQ,LK
      COMMON/SIZESS/NETSS,NDTSS,NIDSS
      COMMON/BEGSS/IBGSS(6)
      COMMON/MTLPRM/SMU,ETA
      COMMON/KI/BCRK(2,2,18)
      EQUIVALENCE (RFF(1),IFF(1)),(REAL(1),INTGR(1)),(IBEGIN(1),ICON),
```

```
     @(IEND(1),LCON),(ISZ(1),NET),(ISZSS(1),NETSS),(RRNG(1),IRNG(1))     LPAN0037
      DATA NSZFF/10000/,NSZRNG/12000/,NSZMN/8000/                        LPAN0038
      DATA PI/3.141593/,S/1.0,3*0.0,1.0,4*0.0/,C/1.0,3*0.0,1.0,4*0.0/    LPAN0039
      KR=5                                                               LPAN0040
      KW=6                                                               LPAN0041
      DTHETA=PI/12.0                                                     LPAN0042
      SQRTPI=SQRT(PI)                                                    LPAN0043
C   INPUT DATA:                                                          LPAN0044
      READ(KR,5000) WIDTH,LENGTH,THK,STFFCT,PRESS,RI,IOFFST              LPAN0045
 5000 FORMAT(6E10.3,I5)                                                  LPAN0046
      READ(KR,5001) A(1),A(2),IPOS(1),IPOS(2)                            LPAN0047
 5001 FORMAT(2E10.3,2I5)                                                 LPAN0048
      READ(KR,5002) E,GNU                                                LPAN0049
 5002 FORMAT(2E10.3)                                                     LPAN0050
      READ(KR,5003) KT1,KT2,KT3                                          LPAN0051
 5003 FORMAT(3I5)                                                        LPAN0052
      ANGLE(1,1)=(IPOS(1)-1)*15.0                                        LPAN0053
      ANGLE(1,2)=(IPOS(2)-1)*15.0                                        LPAN0054
      ANGLE(2,1)=ANGLE(1,1)+180.0                                        LPAN0055
      ANGLE(2,2)=ANGLE(1,2)+180.0                                        LPAN0056
      WRITE(KW,6000) WIDTH,LENGTH,THK,STFFCT,PRESS,RI,IOFFST             LPAN0057
 6000 FORMAT(1H1,58X,14HCONTRACT 81739,//57X,13HPROBLEM NO. 3,///,55X,19LPAN0058
     @HLEFT SIDE STIFFENED,//,60X,11H INPUT DATA,///,16H PLATE GEOMETRY:LPAN0059
     @,//,13H PLATE WIDTH=,E12.5,/,14H PLATE LENGTH=,E12.5,/,17H PLATE TLPAN0060
     @HICKNESS=,E12.5,/,24H PLATE STIFFENER FACTOR=,F8.3,/,25H FAR FIELDLPAN0061
     @ LOADING (PSI)=,E12.5,/,16H RADIUS OF HOLE=,E12.5,/,23H HOLE OFFSELPAN0062
     @T INDICATOR=,I3,27H(=0, HOLE REMAINS CENTERED),//,12H CRACK DATA:,LPAN0063
     @/)                                                                 LPAN0064
      NCRK=0                                                             LPAN0065
      DO 10 I=1,2                                                        LPAN0066
 10   IF( A(I) .GT. 0.0) NCRK=NCRK+1                                     LPAN0067
      DO 15 I=1,NCRK                                                     LPAN0068
 15   WRITE(KW,6001) I,A(I),(IPOS(J),ANGLE(I,J),J=1,2)                   LPAN0069
 6001 FORMAT(11H CRACK NO. ,I3,17H :  CRACK LENGTH=,E12.5,/,18X,23HINITILPAN0070
     @AL CRACK POSITION=,I3,1H(,F8.3,9H DEGREES),/,20X,21HFINAL CRACK POLPAN0071
     @SITION=,I3,1H(,F8.3,9H DEGREES),/)                                 LPAN0072
```

```
      WRITE(KW,6002) E,GNU                                              LPAN0073
 6002 FORMAT(21HOMATERIAL PROPERTIES:,//,20H YOUNGS MODULUS (E)=,E12.5,/LPAN0074
     *,16H POISSONS RATIO=,F12.5,///)                                   LPAN0075
C  CALCULATE S&C MATRICES:                                              LPAN0076
      C(2,1)=GNU                                                        LPAN0077
      C(3,3)=(1.0-GNU)/2.0                                              LPAN0078
      S(2,1)=-GNU                                                       LPAN0079
      S(3,3)=(1.0+GNU)*2.0                                              LPAN0080
      SMU=E/(2.0*(1.0+GNU))                                             LPAN0081
      ETA=(3.0-GNU)/(1.0+GNU)                                           LPAN0082
      GNU=E/(1.0-GNU*GNU)                                               LPAN0083
      DO 20 I=1,3                                                       LPAN0084
      DO 20 J=1,I                                                       LPAN0085
      S(I,J)=S(I,J)/E                                                   LPAN0086
      S(J,I)=S(I,J)                                                     LPAN0087
      C(I,J)=C(I,J)*GNU                                                 LPAN0088
 20   C(J,I)=C(I,J)                                                     LPAN0089
C  DETERMINE NUMBER OF QUADRILATERALS NEEDED ACROSS WIDTH:              LPAN0090
      NELW=WIDTH/(4.0*RI)                                               LPAN0091
C  INSURE THAT NELW IS EVEN VALUED:                                     LPAN0092
      I=NELW/2                                                          LPAN0093
      I=I*2                                                             LPAN0094
      IF(NELW .NE. I ) NELW=NELW-1                                      LPAN0095
      W=WIDTH/NELW                                                      LPAN0096
      RATIO=W/RI                                                        LPAN0097
C  DETERMINE STIFFENER THICKNESS:                                       LPAN0098
      STFTHK=THK+STFFCT*WIDTH*THK/W                                     LPAN0099
C  CENTER NODE REFERENCE (INPUT FOR SUBROUTINE CZONE):                  LPAN0100
      CNODE=NELW/2+1                                                    LPAN0101
      LIMIT=2                                                           LPAN0102
      IF(IOFFST .EQ. 1 ) LIMIT=NELW                                     LPAN0103
      IF( STFFCT .NE. 0 .AND. IOFFST .EQ. 1) LIMIT=LIMIT-1              LPAN0104
C  HOLE WILL INITIALLY BE PLACED IN CENTER OF PLATE.   IF IOFFST IS NOT LPAN0105
C  EQUAL TO 0 THE HOLE CENTER WILL BE MOVED TO THE RIGHT IN INCREMENTS OLPAN0106
C  W AS FAR AS IS POSSIBLE                                              LPAN0107
C                                                                       LPAN0108
```

```
C  PRINT OUT PROGRAM STATISTICS TO DATE:                                  LPAN0109
      IF(KT1 .EQ. KW) GO TO 30                                            LPAN0110
      WRITE(KW,6003) NELW,W,RATIO,CNODE,STFTHK,SMU,ETA                    LPAN0111
 6003 FORMAT(20H0PROGRAM STATISTICS:,//,33H NUMBER OF ELEMENTS ACROSS WILPAN0112
     @DTH=,I4,/,27H QUADRILATERAL DIMENSION W=,E12.5,/,30H HOLE ELEMENT LPAN0113
     @DIMENSION RATIO=,F8.3,/,23H CENTER NODE REFERENCE=,I4,/,21H STIFFELPAN0114
     @NER THICKNESS=,E12.5,//,21H MATERIAL PARAMETERS:,/,5H SMU=,E12.5,/LPAN0115
     @,5H ETA=,E12.5,//,10H S MATRIX:,/)                                  LPAN0116
      WRITE(KW,6004) ((S(I,J),J=1,3),I=1,3)                               LPAN0117
 6004 FORMAT(1H ,3E12.5)                                                  LPAN0118
      WRITE(KW,6005)                                                      LPAN0119
 6005 FORMAT(10H0C MATRIX:)                                               LPAN0120
      WRITE(KW,6004) ((C(I,J),J=1,3),I=1,3)                               LPAN0121
 30   CONTINUE                                                            LPAN0122
      E=0.0                                                               LPAN0123
C  OBTAIN FAR FIELD SUBSTRUCTURE (CHESIRE CAT):                           LPAN0124
 32   CONTINUE                                                            LPAN0125
      PRNTSV=KT1                                                          LPAN0126
      KT1=KT2                                                             LPAN0127
      RO=(1.0+DTHETA)**4*RI                                               LPAN0128
      INDCTR=0                                                            LPAN0129
      CALL FARFLD(WIDTH,LENGTH,NELW,THK,STFTHK,C,S,CNODE,RO,PRESS,INDCTRLPAN0130
     @,NSZFF,RFF,IFF,NSZMN,REAL,INTGR)                                    LPAN0131
      JLOC=IPOS(1)                                                        LPAN0132
      KT1=PRNTSV                                                          LPAN0133
      DO 35 I=1,3                                                         LPAN0134
 35   ISZFF(I)=ISZSS(I)                                                   LPAN0135
      DO 36 I=1,6                                                         LPAN0136
 36   IBGFF(I)=IBGSS(I)                                                   LPAN0137
C  SET UP CRACK PROBLEM:                                                  LPAN0138
      NET=2                                                               LPAN0139
      NDT=96+2*NCRK                                                       LPAN0140
      IPNTR=2*NCRK+146                                                    LPAN0141
      CALL SETUP(NSZMN,1,IPNTR,REAL,INTGR)                                LPAN0142
C  SET UP ASSEMBLY LIST:                                                  LPAN0143
C  ELEMENT NO. 1 IS THE FAR FIELD SUBSTRUCTURE (CHESIRE CAT)              LPAN0144
```

```
      IPNTR=IMASTR+2                                                    LPAN0145
      INTGR(IMASTR)=IPNTR                                               LPAN0146
      DO 40 I=1,48                                                      LPAN0147
      INTGR(IPNTR)=I                                                    LPAN0148
   40 IPNTR=IPNTR+1                                                     LPAN0149
C  ELEMENT NO. 2 IS THE CRACKED RING                                    LPAN0150
      INTGR(IMASTR+1)=IPNTR                                             LPAN0151
   45 IPNTR=INTGR(IMASTR+1)                                             LPAN0152
      DO 50 I=49,NDT                                                    LPAN0153
      INTGR(IPNTR)=I                                                    LPAN0154
   50 IPNTR=IPNTR+1                                                     LPAN0155
      J=(4+JLOC)*2-1                                                    LPAN0156
      IF(JLOC .GE. 21) J=(JLOC-20)*2-1                                  LPAN0157
      DO 60 I=J,48                                                      LPAN0158
      INTGR(IPNTR)=I                                                    LPAN0159
   60 IPNTR=IPNTR+1                                                     LPAN0160
      J=J-1                                                             LPAN0161
      IF(JLOC .EQ. 21) GO TO 75                                         LPAN0162
      DO 70 I=1,J                                                       LPAN0163
      INTGR(IPNTR)=I                                                    LPAN0164
   70 IPNTR=IPNTR+1                                                     LPAN0165
   75 CONTINUE                                                          LPAN0166
      IF(KT1 .EQ. KW) GO TO 80                                          LPAN0167
      J=IMASTR+1                                                        LPAN0168
      WRITE(KW,6006) (INTGR(I),I=IMASTR,J)                              LPAN0169
 6006 FORMAT(22H0MASTER ASSEMBLY LIST:,/,59H POINTER FOR FAR FIELD SUBSTLPAN0170
     0RUCTURED ELEMENT (CHESIRE CAT)=,I5,/,26H POINTER FOR CRACKED RING=LPAN0171
     0,I5,/)                                                            LPAN0172
      J=J+1                                                             LPAN0173
      IPNTR=J+47                                                        LPAN0174
      WRITE(KW,6007) (INTGR(I),I=J,IPNTR)                               LPAN0175
 6007 FORMAT(44H0FAR FIELD SUBSTRUCTURED ELEMENT D.O.F.S:    ,16I5,/44X,1LPAN0176
     06I5,/,44X,16I5)                                                   LPAN0177
      IPNTR=IPNTR+1                                                     LPAN0178
      J=IPNTR+95+NCRK*2                                                 LPAN0179
      WRITE(KW,6008) (INTGR(I),I=IPNTR,J)                               LPAN0180
```

```
 6008 FORMAT(25H0CRACKED RING D.O.F.S:    ,16I5,/,25X,16I5,/,25X,16I5,/,2LPAN0181
     *5X,16I5,/,25X,16I5,/,25X,16I5,/,25X,16I5)                         LPAN0182
 80   CALL ORK(NSZMN,REAL,INTGR)                                         LPAN0183
C  ASSEMBLE FAR FIELD SUBSTRUCTURE:                                      LPAN0184
      DO 85 I=1,3                                                        LPAN0185
 85   ISZSS(I)=ISZFF(I)                                                  LPAN0186
      DO 86 I=1,6                                                        LPAN0187
 86   IBGSS(I)=IBGFF(I)                                                  LPAN0188
      CALL ASMSUB(1,RFF,IFF,REAL,INTGR)                                  LPAN0189
C  COMPUTE AND ASSEMBLE CRACKED RING ELEMENT:                            LPAN0190
      THTCRK=(JLOC-1)*DTHETA                                             LPAN0191
      INDCTR=0                                                           LPAN0192
      PRNTSV=KT1                                                         LPAN0193
      KT1=KT3                                                            LPAN0194
      CALL RING(RI,THK,THTCRK,NCRK,A(1),A(2),INDCTR,S,C,NSZRNG,RRNG,     LPAN0195
     *IRNG)                                                              LPAN0196
      KT1=PRNTSV                                                         LPAN0197
      CALL ASMSUB(2,RRNG,IRNG,REAL,INTGR)                                LPAN0198
C OBTAIN SOLUTION:                                                       LPAN0199
      I=1                                                                LPAN0200
      CALL FACT(I,REAL,INTGR)                                            LPAN0201
      CALL SIMULQ(GNU,REAL,INTGR)                                        LPAN0202
C  OBTAIN CRACKED RING INTERIOR DISPLACEMENTS:                           LPAN0203
      CALL QBACK(2,RRNG,IRNG,REAL,INTGR)                                 LPAN0204
      IF(NCRK .EQ. 0 ) GO TO 120                                         LPAN0205
C  OBTAIN KI AND KII:                                                    LPAN0206
      WRITE(KW,6012)                                                     LPAN0207
 6012 FORMAT(1H0)                                                        LPAN0208
      DO 90 I=1,6                                                        LPAN0209
      ISAVE(I)=IBEGIN(I)                                                 LPAN0210
 90   IBEGIN(I)=IBGSS(I)                                                 LPAN0211
      THTCRK=THTCRK*180.0/PI                                             LPAN0212
      DO 110 I=1,NCRK                                                    LPAN0213
      CALL XTRACT(I+18,18,ELQ,RRNG,IRNG)                                 LPAN0214
      DO 100 J=1,2                                                       LPAN0215
      RK(J)=0.0                                                          LPAN0216
```

```
      DO 100  K=1,18                                                    LPAN0217
  100 RK(J)=RK(J)+BCRK(I,J,K)*ELQ(K)                                    LPAN0218
      DO 105 J=1,2                                                      LPAN0219
  105 RK(J)=RK(J)*SQRTPI                                                LPAN0220
      RK(2)=ABS(RK(2))                                                  LPAN0221
      WRITE(KW,6010) I,THTCRK,(RK(J),J=1,2)                             LPAN0222
 6010 FORMAT(10H CRACK NO.,I2,8H  ANGLE=,F8.3,8H     KI=,E12.5,8H    KIILPAN0223
     *=,E12.5)                                                          LPAN0224
  110 THTCRK=THTCRK+180.0                                               LPAN0225
C  INCREMENT CRACK POSITION:                                            LPAN0226
  120 JLOC=JLOC+1                                                       LPAN0227
      DO 125 I=1,6                                                      LPAN0228
  125 IBEGIN(I)=ISAVE(I)                                                LPAN0229
      IF(JLOC .LE. IPOS(2)) GO TO 45                                    LPAN0230
C  INCREMENT HOLE POSITION:                                             LPAN0231
      IF(IOFFST .EQ. 0) GO TO 130                                       LPAN0232
      IF(CNODE .GE. LIMIT .AND. IOFFST .EQ.  1 ) GO TO 130              LPAN0233
      IF(CNODE .LE. LIMIT .AND. IOFFST .EQ. -1 ) GO TO 130              LPAN0234
      E=E-IOFFST*W                                                      LPAN0235
      CNODE=CNODE+IOFFST                                                LPAN0236
      WRITE(KW,6011) E                                                  LPAN0237
 6011 FORMAT(1H1,53X,24H INCREMENT HOLE POSITION,//,54X,13HHOLE OFFSET =LPAN0238
     *E12.5,///)                                                        LPAN0239
      GO TO 32                                                          LPAN0240
  130 CONTINUE                                                          LPAN0241
      STOP                                                              LPAN0242
      END                                                               LPAN0243
```

```
      SUBROUTINE LUG(WIDTH,LENGTH,NELW,THK,STFTHK,C,NSIZE,RSS,ISS)      LUGL0000
C***********************************************************************LUGL0001
C SUBROUTINE LUG GENERATES A RECTANGULAR REGION WHICH IS USED BY        LUGL0002
C SUBROUTINES FARFLD, SBL AND DBL                                       LUGL0003
C***********************************************************************LUGL0004
C COPYRIGHT (C) 1975 MASSACHUSETTS INSTITUTE OF TECHNOLOGY              LUGL0005
C AEROELASTIC AND STRUCTURES RESEARCH LABORATORY                        LUGL0006
C***********************************************************************LUGL0007
C FINITE ELEMENT APPLICATIONS TO USAF STRUCTURAL INTEGRITY PROBLEMS     LUGL0008
C  LEFT SIDE STIFFENED ONLY                                             LUGL0009
      DIMENSION RSS(1),ISS(1),C(3,3),ELK(36),ELKSTF(36)                 LUGL0010
     *,ANGLE(4),TEMP(4),COORD(12),ND(4),ELQ(8)                          LUGL0011
      INTEGER ELNO                                                      LUGL0012
      REAL LENGTH,L                                                     LUGL0013
      COMMON/IO/KR,KW,KP,KT1,KT2,KT3                                    LUGL0014
      COMMON/SIZE/NET,NDT                                               LUGL0015
      COMMON/BEGIN/IBEGIN(6)                                            LUGL0016
      COMMON/END/IEND(6)                                                LUGL0017
      COMMON/SIZESS/NETSS,NDTSS,NIDSS                                   LUGL0018
      COMMON/BEGSS/IBGSS(6)                                             LUGL0019
      COMMON/STRSS/ B(3,9),BSTF(3,9)                                    LUGL0020
      DATA TEMP/4*0.0/                                                  LUGL0021
      DO 5 I=1,12                                                       LUGL0022
    5 COORD(I)=0.0                                                      LUGL0023
      W=WIDTH/NELW                                                      LUGL0024
      AR=5.0                                                            LUGL0025
      NELL=LENGTH/(W*AR)+1                                              LUGL0026
      NDW=NELW+1                                                        LUGL0027
      NDL=NELL+1                                                        LUGL0028
      NET=NELL*NELW                                                     LUGL0029
      NDT=(NDL+1)*NDW*2                                                 LUGL0030
      NIDSS=NDT-4*NDW                                                   LUGL0031
      IMASTR=NET*9                                                      LUGL0032
      L=LENGTH/NELL                                                     LUGL0033
      AR=L/W                                                            LUGL0034
      COORD(1)=W                                                        LUGL0035
```

```
      COORD(2)=L                                                        LUGL0036
      COORD(5)=L                                                        LUGL0037
      COORD(10)=W                                                       LUGL0038
      LMASTR=NDW*2                                                      LUGL0039
      IF( KT1 .EQ. KW ) GO TO 10                                        LUGL0040
      WRITE(KW,6000)WIDTH,LENGTH,THK,STFTHK,NELW                        LUGL0041
 6000 FORMAT(1H1,60X,9HENTRY LUG,//,23H SUBROUTINE INPUT DATA:,//,11H LULUGL0042
     $G WIDTH=,E12.5,/,12H LUG LENGTH=,E12.5,/,15H LUG THICKNESS=,E12.5,LUGL0043
     $/,25H LUG STIFFENER THICKNESS=,E12.5,30H    ( LEFT SIDE STIFFENED OLUGL0044
     $NLY),/,33H NUMBER OF ELEMENTS ACROSS WIDTH=,I3,/)                 LUGL0045
      WRITE(KW,6001)NELL,NDW,NDL,NET,NDT,NIDSS,(COORD(I),I=1,12),AR     LUGL0046
 6001 FORMAT(27H0SUBROUTINE LUG STATISTICS:,//,37H NUMBER OF ELEMENTS ALLUGL0047
     $ONG LUG LENGTH=,I3,/,29H NUMBER OF NODES ALONG WIDTH=,I3,/,30H NUMLUGL0048
     $BER OF NODES ALONG LENGTH=,I3,/,23H TOTAL NUMBER ELEMENTS=,I3,/,25LUGL0049
     $H TOTAL NUMBER OF D.O.F.S=,/,63H NUMBER OF D.O.F.S TO BE CONDENSEDLUGL0050
     $ OUT GLOBAL STIFFNESS MATRIX=,I3,/,21H ELEMENT COORDINATES:,12F8.3LUGL0051
     $,/,22H ELEMENT ASPECT RATIO=,F8.3)                                LUGL0052
 10   CALL SETUP(NSIZE,LMASTR,IMASTR,RSS,ISS)                           LUGL0053
      ELNO=0                                                            LUGL0054
      IMASTR=IBEGIN(4)                                                  LUGL0055
      DO 30 I=1,NELL                                                    LUGL0056
      INODE1=(I-1)*NDW+1                                                LUGL0057
      LNODE1=INODE1+NELW-1                                              LUGL0058
      DO 20 II=INODE1,LNODE1                                            LUGL0059
      IPNTR=IMASTR+NET+ELNO*8                                           LUGL0060
      ISS(IMASTR+ELNO)=IPNTR                                            LUGL0061
      ELNO=ELNO+1                                                       LUGL0062
      ISS(IPNTR+1)=II*2                                                 LUGL0063
      ISS(IPNTR)=ISS(IPNTR+1)-1                                         LUGL0064
      ISS(IPNTR+2)=ISS(IPNTR+1)+1                                       LUGL0065
 20   ISS(IPNTR+3)=ISS(IPNTR+2)+1                                       LUGL0066
      LNODE1=INODE1+NDW+NELW                                            LUGL0067
      DO 30 II=1,NELW                                                   LUGL0068
      ISS(IPNTR+5)=LNODE1*2                                             LUGL0069
      ISS(IPNTR+4)=ISS(IPNTR+5)-1                                       LUGL0070
      ISS(IPNTR+6)=ISS(IPNTR+4)-2                                       LUGL0071
```

```
      ISS(IPNTR+7)=ISS(IPNTR+6)+1                                        LUGL0072
      IPNTR=IPNTR-8                                                      LUGL0073
30    LNODE1=LNODE1-1                                                    LUGL0074
      IMASTR=IBEGIN(4)      +NET                                         LUGL0075
      LMASTR=IMASTR+NELW*8                                               LUGL0076
      INODE1=NOT-2*NDW+1                                                 LUGL0077
      LNODE1=1                                                           LUGL0078
      DO 35 I=INODE1,NOT                                                 LUGL0079
      DO 34 II=IMASTR,LMASTR                                             LUGL0080
      IF( ISS(II) .EQ. LNODE1 ) ISS(II)=I                                LUGL0081
34    CONTINUE                                                           LUGL0082
35    LNODE1=LNODE1+1                                                    LUGL0083
36    IF(KT1 .EQ. KW ) GO TO 60                                          LUGL0084
      WRITE(6,6002)                                                      LUGL0085
6002  FORMAT(22H0MASTER ASSEMBLY LIST:,//,18H ELEMENT POINTERS:,/)      LUGL0086
      IMASTR=IBEGIN(4)                                                   LUGL0087
      LMASTR=IMASTR+NET-1                                                LUGL0088
      ELNO=0                                                             LUGL0089
      DO 40 I=IMASTR,LMASTR                                              LUGL0090
      ELNO=ELNO+1                                                        LUGL0091
40    WRITE(6,6003)ELNO,ISS(I)                                           LUGL0092
6003  FORMAT(13H ELEMENT NO.=,I5,13H     POINTER=,I5)                    LUGL0093
      IMASTR=LMASTR+1                                                    LUGL0094
      WRITE(6,6004)                                                      LUGL0095
6004  FORMAT(17H0ELEMENT D.O.F.S:,/)                                     LUGL0096
      DO 50 I=1,NET                                                      LUGL0097
      LMASTR=IMASTR+7                                                    LUGL0098
      WRITE(6,6005)I,(ISS(J),J=IMASTR,LMASTR)                            LUGL0099
6005  FORMAT(12H ELEMENT NO.,I3,13H      D.O.F.S:,8I5)                   LUGL0100
50    IMASTR= IMASTR+8                                                   LUGL0101
60    CALL ORK(NSIZE,RSS,ISS)                                            LUGL0102
      CALL QUAD4(COORD,THK,TEMP,C,0,ND,ANGLE,ELQ,ELK,B,1,KW)             LUGL0103
      AR=STFTHK/THK                                                      LUGL0104
      DO 70 I=1,36                                                       LUGL0105
70    ELKSTF(I)=ELK(I)*AR                                                LUGL0106
      DO 80 I=1,3                                                        LUGL0107
```

```
      DO 80 J=1,9                                              LUGL0108
80    BSTF(I,J)=B(I,J)*AR                                      LUGL0109
      NELWM1=NELW-1                                            LUGL0110
      ELNO=0                                                   LUGL0111
      DO 100 IROW=1,NELL                                       LUGL0112
      DO 90 IHERE=1,NELWM1                                     LUGL0113
      ELNO=ELNO+1                                              LUGL0114
90    CALL ASMLTV(ELNO,8,ELK,ELQ,RSS,ISS)                      LUGL0115
      ELNO=ELNO+1                                              LUGL0116
 100  CALL ASMLTV(ELNO,8,ELKSTF,ELQ,RSS,ISS)                   LUGL0117
      LMASTR=NDW*2                                             LUGL0118
      IMASTR=IBEGIN(5)-1                                       LUGL0119
      DO 110 I=1,LMASTR                                        LUGL0120
      ISS(I)=I                                                 LUGL0121
110   RSS(IMASTR+I)=0.0                                        LUGL0122
      CALL BCON(RSS,ISS)                                       LUGL0123
115   I=1                                                      LUGL0124
      CALL STACON(I,NIDSS,RSS,ISS)                             LUGL0125
      DO 120 I=1,6                                             LUGL0126
120   IBGSS(I)=IBEGIN(I)                                       LUGL0127
      NETSS=NET                                                LUGL0128
      NDTSS=NDT                                                LUGL0129
      IF( KT1 .EQ. KW) GO TO 130                               LUGL0130
      WRITE(*,6006)                                            LUGL0131
6006  FORMAT(1H0,60X,8HEXIT LUG,//)                            LUGL0132
130   CONTINUE                                                 LUGL0133
      RETURN                                                   LUGL0134
      END                                                      LUGL0135
```

```
      SUBROUTINE CZONE(WIDTH,NELW,THK,STFTHK,C,S,CNODE,RI,NSIZE,RCZONE, CZNL0000
     *ICZONE,RHOLE,IHOLE)                                               CZNL0001
C*********************************************************************CZNL0002
C  LEFT SIDE STIFFENED                                                 CZNL0003
C SUBROUTINE CZONE GENERATES THE CENTRAL STRIP THAT CONTAINS THE HOLE  CZNL0004
C FOR USE IN THE FARFIELD MESH GENERATED BY SUBROUTINE FARFLD          CZNL0005
C*********************************************************************CZNL0006
C COPYRIGHT (C) 1975 MASSACHUSETTS INSTITUTE OF TECHNOLOGY             CZNL0007
C AEROELASTIC AND STRUCTURES RESEARCH LABORATORY                       CZNL0008
C*********************************************************************CZNL0009
C FINITE ELEMENT APPLICATIONS TO USAF STRUCTURAL INTEGRITY PROBLEMS    CZNL0010
      DIMENSION RCZONE(1),ICZONE(1),RHOLE(1),IHOLE(1),C(3,3),S(3,3),ELK(CZNL0011
     *36),ELKSTF(36),ANGLE(4),TEMP(4),ND(4),COORD(12),ELO(8),NOTASM(4), CZNL0012
     *IELSTF(2),IDOF(8),RHOLE(6,3,13)                                   CZNL0013
      INTEGER CNODE,ELNO                                                CZNL0014
      COMMON/IO/KR,KW,KP,KT1,KT2,KT3                                    CZNL0015
      COMMON/SIZE/NET,NOT                                               CZNL0016
      COMMON/BEGIN/IBEGIN(6)                                            CZNL0017
      COMMON/END/IEND(6)                                                CZNL0018
      COMMON/SIZESS/ISZ(3)                                              CZNL0019
      COMMON/BEGSS/IBGSS(6)                                             CZNL0020
      COMMON/STRSS/R(3,9),BSTF(3,9)                                     CZNL0021
      DATA TEMP/4*0.0/                                                  CZNL0022
      DO 5 I=1,12                                                       CZNL0023
    5 COORD(I)=0.0                                                      CZNL0024
      W=WIDTH/NELW                                                      CZNL0025
      NOW=NELW+1                                                        CZNL0026
      IF( CNODE .GT. 1 .AND. CNODE .LT. NOW ) GO TO 4                   CZNL0027
      WRITE(KW,6008) CNODE,NOW                                          CZNL0028
 6008 FORMAT(60H0***** PROGRAM INTERUPT INITIATED BY SUBROUTINE CZONE **CZNL0029
     ****,///,34H CENTER NODE REFERENCE PARAMETER =,I4,35H IS OUT OF RACZNL0030
     *NGE OF E WIDTH (NELW),,/,44H CNODE MUST BE GREATER THAN 1 AND LESSCZNL0031
     * THAN ,I4,//,53H ***** EXECUTION TERMINATED IN SUBROUTINE CZONE **CZNL0032
     ****)                                                              CZNL032A
      STOP                                                              CZNL0033
    4 NET=1+2*NELW                                                      CZNL0034
      NOT=2*(3*NOW+24)                                                  CZNL0035
```

```
      NIDSS=NDT-4*NDW-48                                                CZNL0036
      NOTASM(1)=CNODE-1                                                 CZNL0037
      NOTASM(2)=CNODE                                                   CZNL0038
      NOTASM(3)=CNODE-1+NELW                                            CZNL0039
      NOTASM(4)=CNODE+NELW                                              CZNL0040
      IELSTF(1)=NELW                                                    CZNL0041
      IELSTF(2)=2*NELW                                                  CZNL0042
      COORD(1)=W                                                        CZNL0043
      COORD(2)=W                                                        CZNL0044
      COORD(5)=W                                                        CZNL0045
      COORD(10)=W                                                       CZNL0046
      LMASTR=(NET-1)*9+65                                               CZNL0047
      IF(KT1 .EQ. KW) GO TO 10                                          CZNL0048
      WRITE(KW,6000) WIDTH,NELW,THK,STFTHK,CNODE,RI                     CZNL0049
 6000 FORMAT(1H1,49X,11HENTRY CZONE,///,23H SUBROUTINE INPUT DATA:,//,13CZNL0050
     *H PLATE WIDTH=,E12.5,/,33H NUMBER OF ELEMENTS ACROSS WIDTH=,I4,/,1CZNL0051
     *7H PLATE THICKNESS=,E12.5,/,21H STIFFENER THICKNESS=,F12.5,/,28H HCZNL0052
     *OLE CENTER NODE REFERENCE=,I3,/,27H HOLE ELEMENT INPUT RADIUS=,E12CZNL0053
     *.5)                                                               CZNL0054
      WRITE(KW,6001) NOW,NIDSS,(NOTASM(I),I=1,4),(IELSTF(I),I=1,2),(COORCZNL0055
     *D(I),I=1,12)                                                      CZNL0056
 6001 FORMAT(29H0SUBROUTINE CZONE STATISTICS:,//,30H NUMBER OF NODES ACRCZNL0057
     *OSS WIDTH=,I4,/,75H NUMBER OF INTERNAL D.O.F.S TO BE CONDENSED OUTCZNL0058
     * OF GLOBAL STIFFNESS MATRIX=,I4,/,30H ELEMENTS NOT TO BE ASSEMBLEDCZNL0059
     *:,4I5,/,26H ELEMENTS TO BE STIFFENED:,4I5,/,21H ELEMENT COORDINATECZNL0060
     *S:,12F8.3)                                                        CZNL060A
   10 CALL SETUP(NSIZE,2,LMASTR,RCZONE,ICZONE)                          CZNL0061
      ELNO=0                                                            CZNL0062
      IMASTR=IBEGIN(4)                                                  CZNL0063
      INODE=NDW+1                                                       CZNL0064
      DO 30 IROW=1,2                                                    CZNL0065
      IF(IROW .EQ. 2) INODE=1                                           CZNL0066
      LNODE=INODE+NELW-1                                                CZNL0067
      DO 20 I=INODE,LNODE                                               CZNL0068
      IPNTR=IMASTR+NET+ELNO*8                                           CZNL0069
      ICZONE(IMASTR+ELNO)=IPNTR                                         CZNL0070
      ELNO=ELNO+1                                                       CZNL0071
```

```
      ICZONE(IPNTR+1)=I*2                                               CZNL0072
      ICZONE(IPNTR)=ICZONE(IPNTR+1)-1                                   CZNL0073
      ICZONE(IPNTR+2)=ICZONE(IPNTR+1)+1                                 CZNL0074
   20 ICZONE(IPNTR+3)=ICZONE(IPNTR+2)+1                                 CZNL0075
      LNODE=NDW                                                         CZNL0076
      IF(IROW .EQ. 2) LNODE=3*NDW                                       CZNL0077
      DO 30 I=1,NFLW                                                    CZNL0078
      ICZONE(IPNTR+5)=LNODE*2                                           CZNL0079
      ICZONE(IPNTR+4)=ICZONE(IPNTR+5)-1                                 CZNL0080
      ICZONE(IPNTR+6)=ICZONE(IPNTR+4)-2                                 CZNL0081
      ICZONE(IPNTR+7)=ICZONE(IPNTR+6)+1                                 CZNL0082
      IPNTR=IPNTR-8                                                     CZNL0083
   30 LNODE=LNODE-1                                                     CZNL0084
      IDOF(1)=CNODE-1+2*NDW                                             CZNL0085
      IDOF(2)=CNODE-1                                                   CZNL0086
      IDOF(3)=CNODE-1+NDW                                               CZNL0087
      IDOF(4)=CNODE+NDW                                                 CZNL0088
      IDOF(5)=CNODE+1+NDW                                               CZNL0089
      IDOF(6)=CNODE+1                                                   CZNL0090
      IDOF(7)=CNODE+1+2*NDW                                             CZNL0091
      IDOF(8)=CNODE+2*NDW                                               CZNL0092
      IPNTR=IMASTR+NET+16*NFLW                                          CZNL0093
      ICZONE(IMASTR+NET-1)=IPNTR                                        CZNL0094
      INODE=1                                                           CZNL0095
      DO 40 I=1,8                                                       CZNL0096
      ICZONE(IPNTR+INODE)=IDOF(I)*2                                     CZNL0097
      ICZONE(IPNTR+INODE-1)=ICZONE(IPNTR+INODE)-1                       CZNL0098
   40 INODE=INODE+2                                                     CZNL0099
      IPNTR=IPNTR+16                                                    CZNL0100
      INODE=2*(3*NDW+1)-1                                               CZNL0101
      LNODE=INODE+47                                                    CZNL0102
      DO 50 I=INODE,LNODE                                               CZNL0103
      ICZONE(IPNTR)=I                                                   CZNL0104
   50 IPNTR=IPNTR+1                                                     CZNL0105
      IF(KT1 .EQ. KW) GO TO 80                                          CZNL0106
      WRITE(KW,6002)                                                    CZNL0107
```

```
6002 FORMAT(22H0MASTER ASSEMBLY LIST:,//,18H ELEMENT POINTERS:,/)         CZNL0108
     IMASTR=IBEGIN(4)                                                     CZNL0109
     LMASTR=IMASTR+NET-1                                                  CZNL0110
     ELNO=0                                                               CZNL0111
     DO 60 I=IMASTR,LMASTR                                                CZNL0112
     ELNO=ELNO+1                                                          CZNL0113
60   WRITE(KW,6003) ELNO,ICZONE(I)                                        CZNL0114
6003 FORMAT(13H ELEMENT NO.=,I5,13H     POINTER=,I5)                      CZNL0115
     IMASTR=LMASTR+1                                                      CZNL0116
     WRITE(KW,6004)                                                       CZNL0117
6004 FORMAT(17H0ELEMENT D.O.F.S:,/)                                       CZNL0118
     IPNTR=NET-1                                                          CZNL0119
     DO 70 I=1,IPNTR                                                      CZNL0120
     LMASTR=IMASTR+7                                                      CZNL0121
     WRITE(KW,6005) I,(ICZONE(J),J=IMASTR,LMASTR)                         CZNL0122
6005 FORMAT(12H ELEMENT NO.,I5,13H     D.O.F.S:,8I5)                      CZNL0123
70   IMASTR=IMASTR+8                                                      CZNL0124
     LMASTR=IEND(4)                                                       CZNL0125
     IMASTR=LMASTR-63                                                     CZNL0126
     WRITE(KW,6006) (ICZONE(J),J=IMASTR,LMASTR)                           CZNL0127
6006 FORMAT(22H0HOLE ELEMENT D.O.F.S:,/,8(1H ,21X,8I5,/))                 CZNL0128
80   CALL ORK(NSIZE,RCZONE,ICZONE)                                        CZNL0129
     CALL QUAD4(COORD,THK,TEMP,C,0,ND,ANGLE,ELQ,ELK,B,1,KW)               CZNL0130
     RATIO=STFTHK/THK                                                     CZNL0131
     DO 90 I=1,36                                                         CZNL0132
90   ELKSTF(I)=ELK(I)*RATIO                                               CZNL0133
     DO 100 I=1,3                                                         CZNL0134
     DO 100 J=1,9                                                         CZNL0135
100  BSTF(I,J)=B(I,J)*RATIO                                               CZNL0136
     IPNTR=NET-1                                                          CZNL0137
     DO 140 N=1,IPNTR                                                     CZNL0138
     DO 110 I=1,4                                                         CZNL0139
     IF( N .EQ. NOTASM(I)) GO TO 140                                      CZNL0140
110  CONTINUE                                                             CZNL0141
     DO 120 I=1,2                                                         CZNL0142
     IF( N .EQ. IELSTF(I)) GO TO 130                                      CZNL0143
```

```
120   CONTINUE                                                         CZNL0144
      CALL ASMLTV(N,8,ELK,ELQ,RCZONE,ICZONE)                           CZNL0145
      GO TO 140                                                        CZNL0146
130   CALL ASMLTV(N,8,ELKSTF,ELQ,RCZONE,ICZONE)                        CZNL0147
140   CONTINUE                                                         CZNL0148
      DO 150 I=1,12                                                    CZNL0149
150   COORD(I)=0.0                                                     CZNL0150
      COORD(1)=2.0*W                                                   CZNL0151
      COORD(4)=COORD(1)                                                CZNL0152
      COORD(5)=COORD(1)                                                CZNL0153
      COORD(8)=COORD(1)                                                CZNL0154
      IMASTR=KT1                                                       CZNL0155
      KT1=KW                                                           CZNL0156
      CALL HOLEL(COORD,THK,S,RI,RHOLE,IHOLE,BHOLE)                     CZNL0157
      KT1=IMASTR                                                       CZNL0158
      CALL ASMSUB(NET,RHOLE,IHOLE,RCZONE,ICZONE)                       CZNL0159
      ICZONE(1)=CNODE*2                                                CZNL0160
      ICZONE(2)=ICZONE(1)-1                                            CZNL0161
      IMASTR=IBEGIN(5)-1                                               CZNL0162
      DO 160 I=1,2                                                     CZNL0163
      J=IMASTR+ICZONE(I)                                               CZNL0164
160   RCZONE(J)=0.0                                                    CZNL0165
      CALL BCON(RCZONE,ICZONE)                                         CZNL0166
      I=1                                                              CZNL0167
      CALL STACON(I,NIDSS,RCZONE,ICZONE)                               CZNL0168
      DO 170 I=1,6                                                     CZNL0169
170   IBGSS(I)=IBEGIN(I)                                               CZNL0170
      ISZ(1)=NET                                                       CZNL0171
      ISZ(2)=NDT                                                       CZNL0172
      ISZ(3)=NIDSS                                                     CZNL0173
      IF( KT1 .EQ. KW ) GO TO 180                                      CZNL0174
      WRITE(KW,6007)                                                   CZNL0175
6007  FORMAT(1H0,59X,10HEXIT CZONE,//)                                 CZNL0176
180   CONTINUE                                                         CZNL0177
      RETURN                                                           CZNL0178
      END                                                              CZNL0179
```

```
      SUBROUTINE FARFLD(WIDTH,LENGTH,NELW,THK,STFTHK,C,S,CNODE,RI,PRESS,FFDL0000
     @INDCTR,NSIZE,RSS,ISS,NSZWRK,RWORK,IWORK)                          FFDL0001
C***********************************************************************FFDL0002
C LEFT SIDE STIFFENED ONLY                                              FFDL0003
C SUBROUTINE FARFLD GENERATES A FINITE ELEMENT MODEL ( A SUBSTRUCTURE)  FFDL0004
C OF A PLATE WITH ONE HOLE  THAT IS CENTERED IN THE VERTICAL DIRECTION  FFDL0005
C AND IS EITHER CENTERED OR OFF-CENTERED IN THE HORIZONTAL DIRECTION    FFDL0006
C***********************************************************************FFDL0007
C COPYRIGHT (C) 1975 MASSACHUSETTS INSTITUTE OF TECHNOLOGY              FFDL0008
C AEROELASTIC AND STRUCTURES RESEARCH LABORATORY                        FFDL0009
C***********************************************************************FFDL0010
C FINITE ELEMENT APPLICATIONS TO USAF STRUCTURAL INTEGRITY PROBLEMS     FFDL0011
      DIMENSION C(3,3),S(3,3),RSS(1),ISS(1),RWORK(1),IWORK(1),RHOLE(2097FFDL0012
     @),IHOLE(2097),ISAVE1(2),ISAVE2(6),ISAVE3(6),IRGSSC(6),ISZSSC(3),  FFDL0013
     @ISZSS(3),ISZ(2)                                                   FFDL0014
      INTEGER CNODE                                                     FFDL0015
      REAL LENGTH,LNGTH                                                 FFDL0016
      COMMON/IO/KR,KW,KP,KT1,KT2,KT3                                    FFDL0017
      COMMON/SIZE/NET,NDY                                               FFDL0018
      COMMON/BEGIN/IBEGIN(6)                                            FFDL0019
      COMMON/END/IEND(6)                                                FFDL0020
      COMMON/SIZESS/NETSS,NDTSS,NIDSS                                   FFDL0021
      COMMON/BEGSS/IBGSS(6)                                             FFDL0022
      EQUIVALENCE (RHOLE(1),IHOLE(1)),(ISZSS(1),NETSS),(ISZ(1),NET)     FFDL0023
      NET=3                                                             FFDL0024
      NDW=NELW+1                                                        FFDL0025
      NDT=8*NDW+48                                                      FFDL0026
      NIDSS=NDT-48                                                      FFDL0027
      IF(INDCTR .EQ. 1 ) NIDSS=NIDSS-4*NDW                              FFDL0028
      LMASTR=12*NDW+51                                                  FFDL0029
      NCON=  NDW+1                                                      FFDL0030
      W=WIDTH/NELW                                                      FFDL0031
      LNGTH=LENGTH/2.0-W                                                FFDL0032
      E=WIDTH/2.0-(CNODE-1)*W                                           FFDL0033
      IF(KT1 .EQ. KW) GO TO 10                                          FFDL0034
      WRITE(KW,6000) WIDTH,LENGTH,NELW,THK,STFTHK,CNODE,RI,PRESS,INDCTR FFDL0035
```

```
6000 FORMAT(1H1,59X,12HENTRY FARFLD,///,22H SUBROUTINE INPUT DATA,//,13FFDL0036
    &H PLATE WIDTH=,E12.5,/,14H PLATE LENGTH=,E12.5,/,33H NUMBER OF ELEFFDL0037
    &MENTS ACROSS WIDTH=,I5,/,17H PLATE THICKNESS=,E12.5,/,21H STIFFENEFFDL0038
    &R THICKNESS=,E12.5,/,33H HOLE CENTER NODE REFERENCE CODE=,I5,/,27HFFDL0039
    & HOLE ELEMENT INPUT RADIUS=,E12.5,/,26H APPLIED TENSION PRESSURE=,FFDL0040
    &E12.5,/,31H STATIC CONDENSATION INDICATOR=,I4,37H    (INDICATOR=0:FFDL0041
    &CHESIRE CAT PROBLEM;,/,39X,73H INDICATOR=1:PLATE WITH END NODES ANFFDL0042
    &D WITH HOLE-NO LOADS OR B.C. APPLIED))                            FFDL0043
     WRITE(KW,6001) LNGTH,F,NIDSS                                      FFDL0044
6001 FORMAT(23H0SUBROUTINE STATISTICS:,//,12H LUG LENGTH=,E12.5,/,13HHOFFDL0045
    &LE OFFSET=,F12.5,/,75H NUMBER OF INTERNAL D.O.F.S TO BE CONDENSED FFDL0046
    &OUT OF GLOBAL STIFFNESS MATRIX=,I5,//)                            FFDL0047
10   CALL SETUP(NSIZE,NCON,LMASTR,RSS,ISS)                             FFDL0048
     IMASTR=IBEGIN(4)                                                  FFDL0049
     IDOF1=1                                                           FFDL0050
     IDOF2=2*(3*NDW+1)-1                                               FFDL0051
     DO 30 N=1,2                                                       FFDL0052
     IPNTR=IMASTR+NET+(N-1)*4*NDW                                      FFDL0053
     ISS(IMASTR+N-1)=IPNTR                                             FFDL0054
     IF( N .EQ. 2) IDOF1=(2*NDW+1)*2-1                                 FFDL0055
     IF( N .EQ. 2) IDOF2=(NDW+1)*2-1                                   FFDL0056
     LMASTR=IDOF1+2*NDW-1                                              FFDL0057
     DO 20 I=IDOF1,LMASTR                                              FFDL0058
     ISS(IPNTR)=I                                                      FFDL0059
20   IPNTR=IPNTR+1                                                     FFDL0060
     LMASTR=IDOF2+2*NDW-1                                              FFDL0061
     DO 30 I=IDOF2,LMASTR                                              FFDL0062
     ISS(IPNTR)=I                                                      FFDL0063
30   IPNTR=IPNTR+1                                                     FFDL0064
     ISS(IMASTR+2)=IPNTR                                               FFDL0065
     IDOF1=1                                                           FFDL0066
     IDOF2=(NDW+1)*2-1                                                 FFDL0067
     LMASTR=NDW*2                                                      FFDL0068
     DO 40 I=IDOF1,LMASTR                                              FFDL0069
     ISS(IPNTR)=I                                                      FFDL0070
40   IPNTR=IPNTR+1                                                     FFDL0071
```

```
      LMASTR=4*NDW                                                      FFDL0072
      DO 50 I=IDOF2,LMASTR                                              FFDL0073
      ISS(IPNTR)=I                                                      FFDL0074
50    IPNTR=IPNTR+1                                                     FFDL0075
      IDOF1=8*NDW+1                                                     FFDL0076
      IDOF2=IDOF1+47                                                    FFDL0077
      DO 60 I=IDOF1,IDOF2                                               FFDL0078
      ISS(IPNTR)=I                                                      FFDL0079
60    IPNTR=IPNTR+1                                                     FFDL0080
      IF(KTI .EQ. KW) GO TO 80                                          FFDL0081
      IDOF1=IBEGIN(4)                                                   FFDL0082
      IDOF2=IDOF1+2                                                     FFDL0083
      WRITE(KW,6002) (ISS(I),I=IDOF1,IDOF2)                             FFDL0084
6002  FORMAT(22H0MASTER ASSEMBLY LIST:,/,10H POINTERS:,3I5)             FFDL0085
      DO 70 N=1,2                                                       FFDL0086
      IDOF1=IBEGIN(4)+NET+(N-1)*4*NDW                                   FFDL0087
      IDOF2=IDOF1+4*NDW-1                                               FFDL0088
      WRITE(KW,6003) N                                                  FFDL0089
6003  FORMAT(13H0LUG ELEMENT ,I2,9H D.O.F.S:,/)                         FFDL0090
70    WRITE(KW,6004) (ISS(I),I=IDOF1,IDOF2)                             FFDL0091
6004  FORMAT(1H ,20I6)                                                  FFDL0092
      IDOF1=IDOF2+1                                                     FFDL0093
      IDOF2=IEND(4)                                                     FFDL0094
      WRITE(KW,6005)                                                    FFDL0095
6005  FORMAT(23H0CZONE ELEMENT D.O.F.S:,/)                              FFDL0096
      WRITE(KW,6004) (ISS(I),I=IDOF1,IDOF2)                             FFDL0097
80    CALL ORK(NSIZE,RSS,ISS)                                           FFDL0098
      NIDSV=NIDSS                                                       FFDL0099
      DO 90 I=1,2                                                       FFDL0100
90    ISAVE1(I)=ISZ(I)                                                  FFDL0101
      DO 100 I=1,6                                                      FFDL0102
      ISAVE2(I)=IBEGIN(I)                                               FFDL0103
100   ISAVE3(I)=IEND(I)                                                 FFDL0104
      CALL LUG(WIDTH,LNGTH,NELW,THK,STFTHK,C,NSZWRK,RWORK,IWORK)        FFDL0105
      DO 110 I=1,2                                                      FFDL0106
110   ISZ(I)=ISAVE1(I)                                                  FFDL0107
```

```
      DO 120 I=1,6                                                      FFDL0108
      IBEGIN(I)=ISAVE2(I)                                               FFDL0109
  120 IEND(I)=ISAVE3(I)                                                 FFDL0110
      DO 130 N=1,2                                                      FFDL0111
  130 CALL ASMSUB(N,RWORK,IWORK,RSS,ISS)                                FFDL0112
      DO 140 I=1,2                                                      FFDL0113
  140 ISAVE1(I)=ISZ(I)                                                  FFDL0114
      DO 150 I=1,6                                                      FFDL0115
      ISAVE2(I)=IBEGIN(I)                                               FFDL0116
  150 ISAVE3(I)=IEND(I)                                                 FFDL0117
      CALL CZONE(WIDTH,NELW,THK,STFTHK,C,S,CNODE,RI,NSZWRK,RWORK,IWORK, FFDL0118
     @RHOLE,IHOLE)                                                      FFDL0119
      DO 160 I=1,2                                                      FFDL0120
  160 ISZ(I)=ISAVE1(I)                                                  FFDL0121
      DO 170 I=1,6                                                      FFDL0122
      IBEGIN(I)=ISAVE2(I)                                               FFDL0123
      IEND(I)=ISAVE3(I)                                                 FFDL0124
  170 IBGSSC(I)=IBGSS(I)                                                FFDL0125
      DO 180 I=1,3                                                      FFDL0126
  180 ISZSSC(I)=ISZSS(I)                                                FFDL0127
      CALL ASMSUB(3,RWORK,IWORK,RSS,ISS)                                FFDL0128
      NIDSS=NIDSV                                                       FFDL0129
      IF(INDCTR .EQ. 0) GO TO 190                                       FFDL0130
C  PLATE WITH END NODES AND HOLE-NO B.C. OR APPLIED LOADS               FFDL0131
      I=1                                                               FFDL0132
      CALL STACON(I,NIDSS,RSS,ISS)                                      FFDL0133
      GO TO 220                                                         FFDL0134
C  CHESIRE CAT PROBLEM                                                  FFDL0135
C  APPLY B.C.                                                           FFDL0136
  190 IDOF1=(2*NDW+1)*2                                                 FFDL0137
      IDOF2=6*NDW                                                       FFDL0138
      IMASTR=1                                                          FFDL0139
      LMASTR=IBEGIN(5)-1                                                FFDL0140
      DO 200 I=IDOF1,IDOF2,2                                            FFDL0141
      ISS(IMASTR)=I                                                     FFDL0142
      RSS(LMASTR+I)=0.0                                                 FFDL0143
```

```
 200  IMASTR=IMASTR+1                                              FFDL0144
      IDOF1=(2*NDW+1+NDW/2)*2-1                                    FFDL0145
      ISS(IMASTP)=IDOF1                                            FFDL0146
      RSS(LMASTP+IDOF1)=0.0                                        FFDL0147
      IDOF1=6*NDW+2                                                FFDL0148
      IDOF2=8*NDW-2                                                FFDL0149
      F=    W*PRESS*THK/2.0                                        FFDL0150
      DO 210 I=IDOF1,IDOF2,2                                       FFDL0151
      RSS(LMASTR+I)=RSS(LMASTR+I)+F                                FFDL0152
 210  RSS(LMASTR+I+2)=RSS(LMASTR+I+2)+F                            FFDL0153
C ADD IN ADDITIONAL LOADS FOR STIFFENED EDGES                      FFDL0154
      F=(STFTHK-THK)     *W*PRESS/2.0                              FFDL0155
      RSS(LMASTR+IDOF2+2)=RSS(LMASTR+IDOF2+2)+F                    FFDL0156
      RSS(LMASTR+IDOF2  )=RSS(LMASTR+IDOF2  )+F                    FFDL0157
      CALL BCON(RSS,ISS)                                           FFDL0158
      I=1                                                          FFDL0159
      CALL STACON(I,NIDSS,RSS,ISS)                                 FFDL0160
 220  DO 221 I=1,2                                                 FFDL0161
 221  ISZSS(I)=ISZ(I)                                              FFDL0162
      ISZSS(3)=NIDSS                                               FFDL0163
      DO 222 I=1,6                                                 FFDL0164
 222  IBGSS(I)=IBEGIN(I)                                           FFDL0165
      IF( KT1 .EQ. KW ) GO TO 230                                  FFDL0166
      WRITE(KW,6006)                                               FFDL0167
6006  FORMAT(1H0,59X,11HEXIT FARFLD,///)                           FFDL0168
 230  CONTINUF                                                     FFDL0169
      RETURN                                                       FFDL0170
      END                                                          FFDL0171
```